Mining Electronic Health Records: Turning Unstructured Text into Research Data Using Natural Language Processing

Laursen, Martin Sundahl

DOI:
10.21996/ma3z-rf07

Publication date:
2023

Document version:
Final published version

Mining Electronic Health Records: Turning Unstructured Text into Research Data Using Natural Language Processing

A dissertation submitted for the degree of
Doctor of Philosophy

Martin Sundahl Laursen

SDU Robotics
The Maersk Mc-Kinney Moller Institute
The Faculty of Engineering
University of Southern Denmark

Odense, August 2023

Supervisor: Thiusius Rajeeth Savarimuthu

SDU Robotics
The Maersk Mc-Kinney Moller Institute
The Faculty of Engineering
University of Southern Denmark

Campusvej 55
5230 Odense M, Denmark
Phone +45 6550 7380
www.sdu.dk

Co-supervisor: Pernille Just Vinholt

The Department of Clinical Biochemistry
Odense University Hospital

J. B. Winsløws Vej 4
5000 Odense C, Denmark
Phone +45 6611 3333
www.ouh.dk

Abstract

Mining Electronic Health Records: Turning Unstructured Text into Research Data Using Natural Language Processing

The electronic health record (EHR) contains a detailed account of the patients' medical history and is an important source of clinical research data. It is estimated that 80% of the information in the EHR is unstructured and this complicates locating relevant information. The medical history is also registered in a structured format as International Classification of Diseases codes and in medical databases.

Clinical research often relies on coded information for data, but it has some disadvantages: some information from the medical history is not coded; the codes can be erroneous; medical databases only exist for certain patient groups; and the codes are registered after the contact or admission, which introduces a delay. Therefore, some clinical studies rely on manually extracting research data from unstructured EHR text, which is time-consuming and expensive.

Natural language processing (NLP) is a subfield of machine learning and artificial intelligence concerned with the analysis and processing of text and speech. NLP has the potential to automatically extract structured research data from unstructured EHR text. This can significantly reduce the data collection time for studies that currently rely on manual data extraction. Furthermore, NLP can facilitate that studies currently relying on coded data can supplement, improve, or replace the research data with data automatically extracted from the unstructured EHR text.

The overall purpose of this PhD dissertation is to enable Danish clinical research by automatically extracting research data from unstructured EHR text.

The work has focused on bleeding and venous thromboembolism (VTE) as specific parameters for extraction since erroneous coding has been demonstrated for both. I present two deep learning pipelines to extract bleeding and VTE events and their anatomical location from unstructured EHR text.

Furthermore, I present a trained deep learning model to automatically extract the medical history in the form of clinical events, their attributes, and their relations from unstructured EHR text.

Two challenges to using NLP to extract research data are the need for large amounts of annotated data, and that data need to be de-identified to comply with the General Data Protection Regulation. Therefore, I present a deep learning model for de-identification of names, streets, and locations in Danish unstructured EHR text. The model was trained with weak supervision to remove the need for a time-consuming and expensive annotation process. Finally, I present Danish clinical word embeddings that can be used to create a dense clinical annotation corpus and increase the efficiency of the annotation process.

Resumé

Datamining i elektroniske patientjournaler: Fra fritekst til forskningsdata ved brug af sprogteknologi

Den elektroniske patientjournal (EPJ) indeholder en detaljeret redegørelse for patienternes sygehistorie og er en vigtig kilde til klinisk forskningsdata. Det anslås, at 80% af data i EPJ er ustrukturerede, hvilket besværliggør at finde relevant information hurtigt. Sygehistorien er også registreret i et struktureret format via koder fra den Internationale Klassifikation af Sygdomme og i medicinske databaser.

Den kodede information bliver ofte brugt til klinisk forskningsdata, men den har nogle ulemper: Nogle oplysninger fra sygehistorien er ikke kodet; der kan være kodet forkert; medicinske databaser findes kun for visse patientgrupper; og koderne registreres efter kontakten eller indlæggelsen, hvilket medfører en forsinkelse. Derfor afhænger nogle kliniske studier af manuelle udtræk af forskningsdata fra ustruktureret EPJ-tekst, hvilket er tidskrævende og dyrt.

Sprogteknologi er et underområde inden for maskinlæring og kunstig intelligens, der beskæftiger sig med analyse og behandling af tekst og tale. Sprogteknologi har potentialet til automatisk at udtrække strukturerede forskningsdata fra ustruktureret EPJ-tekst. Automatisk udtræk kan reducere dataindsamlingstiden betydeligt for studier, der før var afhængige af manuelt dataudtræk. Sprogteknologi kan desuden understøtte, at studier der før var afhængige af kodede data, kan supplere, forbedre eller erstatte kodet data med data, der automatisk er udtrukket fra den ustrukturerede EPJ-tekst.

Det overordnede formål med denne ph.d.-afhandling er at understøtte dansk klinisk forskning ved automatisk at udtrække forskningsdata fra ustruktureret EPJ-tekst.

Arbejdet har fokuseret på blødning og venøs tromboemboli (VTE) som specifikke parametre for udtræk fra sygehistorien, da fejlkodning er blevet påvist for begge. Jeg præsenterer to deep learning pipelines til at udtrække blødninger og VTE-hændelser og deres anatomiske placering fra ustruktureret EPJ-tekst.

Jeg præsenterer desuden en trænet deep learning-model til automatisk at udtrække sygehistorien i form af kliniske hændelser, deres egenskaber, og relationer fra ustruktureret EPJ-tekst.

To udfordringer, ved at bruge sprogteknologi til at udtrække forskningsdata fra ustruktureret EPJ-tekst, er behovet for store mængder annoterede data, og at data skal anonymiseres for at overholde persondataforordningen. Derfor præsenterer jeg en deep learning-model til anonymisering af dansk ustruktureret EPJ-tekst for navne, gader, og lokationer. Modellen blev trænet med "weak supervision" for at fjerne behovet for en tidskrævende og dyr annoteringsproces. Til sidst præsenterer jeg danske kliniske ordvektorer, der kan bruges til at skabe et kompakt klinisk annoteringskorpus og øge effektiviteten af annoteringsprocessen.

Acknowledgements

I would like to thank Thiusius Rajeeth Savarimuthu for sparking my interest in artificial intelligence when supervising my bachelor thesis and for his continued supervision throughout my master and PhD studies. I have you to thank for all the opportunities this path in my education has given me. I have thoroughly enjoyed both to be educated by you and work with you.

I am very grateful to have had Pernille Just Vinholt as my co-supervisor. You opened the door for me to the extremely interesting world of research and innovation in the healthcare sector. It is always a pleasure and very inspiring to work with you.

It has been a huge pleasure to collaborate with Jannik Skyttegaard Pedersen every day of my PhD studies. Thanks for having made it so much fun to go to work and for all the time we spent discussing both life and work.

Also, thanks to all my other colleagues at SDU Robotics at the University of Southern Denmark and the Department of Clinical Biochemistry at Odense University Hospital—especially to Rasmus Søgaard Hansen, Rasmus Bank Lynggaard, Anne Alnor, and Signe Westerberg Biørn for our collaborations.

Thanks to Cristina Soguero-Ruiz for hosting me at her department during my change of research environment at Universidad Rey Juan Carlos. It was a great time and very inspiring.

I would like to thank all of my family and friends for always supporting me and helping me to balance doing my best without breaking my neck on it—I am grateful for you all. Finally, a special thanks to my girlfriend Andrea for her ability to make me see very clearly the important things in life.

Table of Contents

Abstract **iii**

Resumé **v**

Acknowledgements **vii**

1 Introduction **1**
1.1 Motivation . . . 1
1.1.1 The Electronic Health Record . . . 1
1.1.2 ICD Codes and Medical Databases . . . 2
1.1.3 Sources of Research Data and Their Strengths and Weaknesses . . . 3
1.1.4 When Unstructured Text Is Necessary . . . 4
1.1.5 Compliance with the General Data Protection Regulation . . . 5
1.1.6 Turning Unstructured Text into Research Data . . . 5
1.2 Problem Statement . . . 7
1.3 Contributions . . . 8
1.4 Reading Guide . . . 9

2 A Brief Introduction to Natural Language Processing **11**
2.1 Recurrent Neural Networks . . . 11
2.2 Convolutional Neural Networks . . . 13
2.3 Transformers . . . 14
2.3.1 Architecture . . . 15
2.3.2 Input Embeddings . . . 17
2.3.3 Encoder Block . . . 17

3 Related Work **21**
3.1 Identification of Bleeding Events . . . 21
3.2 Identification of Venous Thromboembolism Events . . . 22
3.3 Extraction of Medical History Parameters . . . 24
3.4 De-identification of Unstructured Electronic Health Record Text . . . 25

4 Identification of Bleeding Events **27**
4.1 Pre-trained Large Language Models 28
4.1.1 Bidirectional Encoder Representations from Transformers 28
4.1.2 Efficiently Learning an Encoder that Classifies Token Replacements Accurately . . . 30
4.2 Classification of Bleeding Events 31
Paper: "Doctors identify hemorrhage better during chart review when assisted by artificial intelligence" 32
4.3 Classification of Bleeding Location 35
Paper: "Domain over size: Clinical ELECTRA surpasses general BERT for bleeding site classification in the free text of electronic health records" 36
4.4 Concluding Remarks 41

5 Identification of Venous Thromboembolism Events **43**
5.1 Pre-trained Large Language Model 44
5.2 Classification of Venous Thromboembolism Events and Locations 44
5.3 Concluding Remarks 45

6 Extraction of Medical History **47**
6.1 Named Entity Recognition and Relation Extraction 48
6.2 Danish Clinical Named Entity Recognition and Relation Extraction 48
Paper: "Danish Clinical Named Entity Recognition and Relation Extraction" 49
6.3 Concluding Remarks 62

7 De-identification of Unstructured Electronic Health Record Text **63**
7.1 Weakly Supervised Model for De-identification of Unstructured EHR Text 64
Paper: "Automatic Annotation of Training Data for Deep Learning Based De-identification of Narrative Clinical Text" 65
7.2 Concluding Remarks 81

8 Faster Annotation of a Dense Corpus **83**
8.1 Creating a Dense Annotation Corpus 83
8.2 Non-contextual Word Embeddings 84
8.3 Training and Benchmarking Danish Clinical Word Embeddings 86
Paper: "Benchmark for evaluation of danish clinical word embeddings" 86
8.4 Concluding Remarks 102

9 Conclusion **103**
9.1 Future Work 105

Bibliography **107**

A MeDa-BERT: A medical Danish pretrained transformer model **119**
Paper: "MeDa-BERT: A medical Danish pretrained transformer model" 119

List of Abbreviations

BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	bidirectional LSTM
BioBERT	biomedical BERT
CBOW	Continuous Bag-of-Words
CNN	convolutional neural network
CNS	central nervous system
CRF	conditional random field
CT	computed tomography
DVT	deep vein thrombosis
EHR	electronic health record
ELECTRA	Efficiently Learning an Encoder that Classifies Token Replacements Accurately
ENT	ear-nose-throat
GDPR	General Data Protection Regulation
GI	gastrointestinal
GloVe	Global Vectors
GPT	generative pre-trained transformer
GRU	gated recurrent unit
HIPAA	Health Insurance Portability and Accountability Act
ICD	International Classification of Diseases
LLM	large language model
LSTM	long short-term memory cell
MD	medical doctor
MLM	masked language modelling
NER	named entity recognition
NLP	natural language processing
NSP	next sentence prediction
PE	pulmonary embolism
PII	personal identifiable information

PURE	Princeton University Relation Extraction system
RE	relation extraction
ReLU	rectified linear unit
RNN	recurrent neural network
RTD	replaced token detection
seq2seq	sequence-to-sequence
SVM	support vector machine
VTE	venous thromboembolism

CHAPTER 1

Introduction

In this chapter, I motivate the work presented in this dissertation and give the problem statement and contributions.

1.1 Motivation

I start by describing the electronic health record (EHR) and how the unstructured text it contains is also coded as International Classification of Diseases (ICD) codes and reported to medical databases. Next, I describe the limitations of using coded data for research and argue that they are insufficient for certain clinical studies that must instead manually extract data from unstructured text, which is time-consuming and expensive. Finally, I argue that natural language processing (NLP) has the potential to replace manual extraction by automatically extracting any relevant information directly from unstructured EHR text.

1.1.1 The Electronic Health Record

The EHR is used to store information about a patient's contacts with the healthcare system and make it available to healthcare staff. Every time a patient is in contact with the healthcare system, relevant information about the contact is noted in the EHR [1].

Healthcare staff such as medical doctors (MDs), nurses, and physiotherapists use the EHR to track the patient's medical history including symptoms, diagnoses, treatments, and test results [2]. The information is used for critical health care tasks such as preparing before patient consultations or surgery, guiding diagnostic decision making, or developing treatment plans.

Information that is not related to the medical history is also noted in the EHR. This includes administrative notes and notes taken during admission about, e.g. visits from family or friends, food preferences, general well-being, quality of sleep, personal hygiene, etc. This information is not directly relevant for the health care of the patient but highly relevant on, e.g. the bed ward to ensure as comfortable an admission as possible for the patient.

Notes in the EHR are presented in chronological order with information about, e.g. department and healthcare staff. Some information such as lab results are stored in structured tables but much of the information in the EHR is unstructured, narrative text that is written or dictated directly by healthcare staff [3]. Unstructured information includes, e.g. clinical notes, radiology descriptions, and pathology answers. It is estimated that 80% of the information in the EHR is unstructured [4]. During an admission, multiple notes may be added per day, and an EHR can consist of thousands of notes. The multiple different sources of information and purposes of use lead to a large amount of diverse information stored as unstructured text in numerous consecutive notes. This complicates locating relevant information in the EHR.

1.1.2 ICD Codes and Medical Databases

Apart from being stored in large part as unstructured text, the medical history is also registered in a structured format as ICD codes and in medical databases.

ICD codes provide a common language for safe and clear communication between healthcare staff, departments, and hospitals. They are used to record, report, and monitor diseases, and the collected statistics are used for, e.g. planning, research, and prevention [5, 6].

The medical databases are broadly divided into administrative registries, health registries that record patients at the time of disease detection, and clinical quality databases that record detailed clinical data on treatment for quality control. The medical databases collect data about multiple variables including diagnoses, procedures, treatments, and lifestyle factors [7, 8]. MDs are required by law to report to medical databases (e.g. [9, 10]).

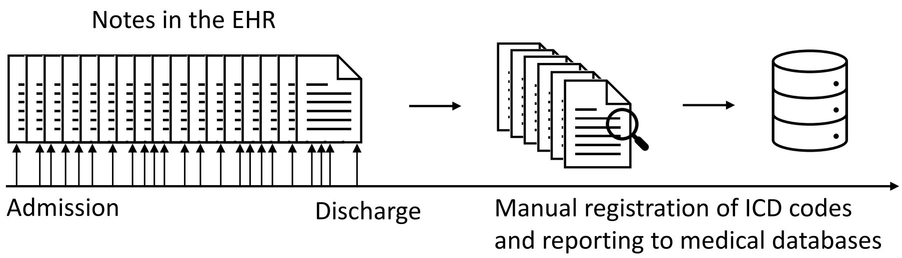

FIGURE 1.1: An overview of the process from admission of a patient to registration of ICD codes and reporting to medical databases. Abbreviations: EHR, electronic health record; ICD, International Classification of Diseases.

After a patient contact, e.g. when a patient is discharged after an admission, ICD codes are coded manually by staff that read through the EHR notes and register the relevant information [7]. ICD codes are also used to report to medical databases along with other information that is gathered manually by reading though the unstructured EHR text. Figure 1.1 shows an overview of the process from admission of a patient to registration of ICD codes and reporting to medical databases.

1.1.3 Sources of Research Data and Their Strengths and Weaknesses

Both unstructured EHR text, and coded information in ICD codes and medical databases can be sources of clinical research data. They have distinct strengths and weaknesses.

Unstructured Text

The unstructured EHR text contains the original and detailed account of patients' medical history. Depending on the type of clinical research, any information from the medical history might be relevant. Therefore, the unstructured text can be of great value as a source of clinical research data. The unstructured nature of the information, though, makes it infeasible to extract research data from the raw EHR data as the text must be read manually to extract the relevant information. This is both time-consuming and expensive as relevant information could span the entire medical history—and therefore also the whole life of the patient [3]. Based on the experience of our research group, we estimate that reading through a single EHR would take one hour, which adds up to an infeasible amount of time for large clinical studies with populations consisting of thousands of patients. The manual extraction limits the number of variables and samples that can be included in the studies and this is reflected in the quality [11].

ICD Codes and Medical Databases

ICD codes and medical databases contain selected structured information on the medical history, which has been registered and reported from the raw unstructured EHR text. Advantages of using medical databases as source of research data include that: they provide opportunity for population-based studies with very large sample sizes, they include nearly all individuals in the target population, and they are collected independently of each research study [7].

Clinical research on ICD codes is limited by the fact that relevant information from the medical history other than the primary diagnoses, e.g. symptoms, lifestyle factors, comorbidities, and secondary diagnoses, are not always coded [12]. Furthermore, ICD codes are known to be unreliable with many studies having documented low sensitivity for several codes [13, 14], and medical databases only exist for certain patient groups and have varying sensitivity and precision [8].

Retrospective epidemiologic research often uses ICD codes and medical databases as data to study causes, effects, and correlation. The described limitations of coded data are often ignored [7] as sample sizes typically are multiple thousands and errors are expected to cancel out in the case and control groups. Insufficient reporting of codes, though, has previously led to the finding of false correlations. Several studies found that diabetes appeared to protect against death in patients with cardiovascular disease. This result was most likely caused by the failure to record less acute diagnoses for patients who were very ill [12, 15]. Selection bias constitutes another problem where studies tend to include variables that are present in medical databases and ignore those that are not. This can lead to, e.g. lifestyle factors, symptoms, and comorbidities not being investigated in epidemiologic studies, and that those variables included may inadvertently appear to be risk factors [15]. Additionally, the fact that codes are registered *after* the contact or admission, which can last for months, can lead to a delay in the discovery of new findings [7].

1.1.4 When Unstructured Text Is Necessary

Clinical research that requires parameters that are not represented in ICD codes or medical databases must rely on manual extraction from unstructured EHR text. Prospective studies also tend to gather data by noting them down or manually extracting them from the unstructured text of the EHR during the study period. Opposite to epidemiologic studies, research on the level of individuals requires such high data accuracies that erroneous codes cannot be used. An example is testing a new treatment in a patient with certain risk factors. Here, it is important that risk factors are known with high accuracy. When codes have poor accuracy, studies can resort to manual extraction from unstructured text. Bleeding and venous thromboembolism (VTE) are examples of two conditions that have poor coding accuracy. Bleeding, VTE, and their coding accuracies are described below.

A bleeding is an escape of blood from a ruptured blood vessel[1]. Bleeding is associated with increased morbidity and mortality [16, 17] and occurs in 3.2% of all medical patients within 14 days of admission. 1.2% of bleedings are considered major bleeding, which can be fatal to the patient [18].

Studies have reported imperfect sensitivity and precision for bleeding in codes [19–27]. Øie et al. found that 8.6% of discharge diagnoses of intracranial bleeding were coded incorrectly [20]. Lain et al. reported that diagnosis and procedure codes for obstetric bleeding tended to be underreported with sensitivities ranging from 28.3% to 100% [25]. Arnason et al. found that ICD codes for bleeding had a sensitivity of 93%, specificity of 88%, and precision of 91%. For major bleeding, they were 94%, 83%, and 87%, respectively [26]. Generally, studies reported low precision with Valkhoff et al. finding a precision of 77% for upper gastrointestinal (GI) bleeding [19] and Andrade et al. finding 9% for GI bleeding [22]. ICD codes for bleeding in patients receiving anticoagulant treatment were reported to have a precision of 58.1% by Delate et al. [21] and 52.5% by Joos et al. [27]. The imperfect sensitivity of bleeding may be caused by it often being a secondary diagnosis, which are not always coded [12].

A VTE is the forming of a blood clot in a vein. It includes deep vein thrombosis (DVT) where a clot is formed in the deep veins, typically in the lower extremities, and pulmonary embolism (PE) where a clot travels through the bloodstream to the lungs [28]. Studies have shown that up to 14.9% of medical patients had a VTE when not treated with thromboprophylaxis [29, 30]. The incidence of VTE is 100 times greater in hospitalised patients than in the general population [31] and is a major cause of morbidity and mortality [29, 32]. One study found 3% and 31% mortality rates for DVT and PE, respectively, compared to 0.4% in the comparison cohort within 30 days of VTE [33]. PE is considered the most common preventable cause of death in hospitalised patients [18].

[1] Definition from Oxford Languages

Studies have consistently reported low sensitivity and precision for VTE in codes and medical databases [26, 34–39]. Arnason et al. reported a sensitivity of 97%, specificity of 74%, and precision of 62% for ICD codes for VTE [26]. Branchford et al. found a sensitivity of 77% for discharge ICD codes for VTE in children [39]. For DVT and PE, Casez et al. reported a sensitivity of 58.0% and 88.9%, respectively [37]. Precision of discharge diagnoses of VTE in the Danish National Patient Registry was found by Severinsen et al. to be 75.0% for diagnoses coded at wards [34]. White et al. found that pregnancy-specific ICD codes for VTE had a precision of 31% and standard VTE had one of 80%. Baumgartner et al. reported a precision of 31.1% for ICD codes to identify recurrent VTE [36]. Fang et al. reported a precision of 64.4% for hospital and emergency department patients and 30.9% for outpatients [38]. The low precision for VTE codes might be due to the failure to remove the code again after an admission where the diagnosis was ruled out [12, 15].

1.1.5 Compliance with the General Data Protection Regulation

When processing personal data, such as that found in the EHR, researchers must comply with the General Data Protection Regulation (GDPR). In brief, some of the main principles of the GDPR for the processing of data are that: they must only be collected for specific purposes; must be limited to what is necessary for the purposes of the processing; must be accurate; must not be kept in a form that can identify the data subjects for longer than necessary for processing; and must be kept safe. While ensuring data privacy and security, the GDPR is a barrier to research due to the demanding data management and the need for explicit consent to use the data [40]. Apart from general clinical research, the need for de-identification also specifically limits clinical NLP research since clinical datasets and pre-trained models cannot be made public if they can leak personal identifiable information (PII) [41].

If data are irreversibly de-identified, they are no longer personal, and do therefore not fall under the GDPR. De-identification includes removal of all PII, which is any information that can identify a person—either directly, or indirectly when combined with other data. PII includes, e.g. names, identification numbers, and location data [40]. De-identification is a particular challenge for unstructured text because of the heterogeneity of the task. Manual de-identification is time-consuming and expensive and this further substantiates the impracticality of using large amounts of unstructured text data for clinical studies. The use of automatic tools for de-identification of unstructured EHR text is limited by the clinical domain often requiring specialised tools due to the specific terms and sentence structures used [42]. Danish being a low-resource language when it comes to available tools and datasets for development of these, is an additional limitation [43].

1.1.6 Turning Unstructured Text into Research Data

NLP is a subfield of machine learning and artificial intelligence concerned with the analysis and processing of text and speech. NLP has the potential to automatically extract structured research data from unstructured EHR text [44].

Automatic extraction can significantly reduce the data collection time for studies that currently rely on manual data extraction. This can potentially facilitate the inclusion of more variables and samples in the studies and thereby improve the quality of them [11]. Automatic extraction can also reduce time spent on reporting data to medical databases.

Furthermore, NLP can facilitate that studies currently relying on coded data can supplement, improve, or replace the research data with data automatically extracted from the unstructured EHR text. This can facilitate large epidemiologic studies on medical history parameters that do not exist in ICD codes or medical databases and thereby decrease selection bias with research parameters. Examples include some symptoms, lifestyle factors, comorbidities, and secondary diagnoses [12].

In this dissertation, the work has focused on bleeding and VTE as specific parameters for extraction from the medical history since erroneous coding has been demonstrated for both. Furthermore, focus has been on automatic extraction of a complete structured medical history, including those parameters not coded or erroneously coded, such as symptoms.

It generally requires large amounts of labelled samples to train NLP models. The annotation process is—like manual extraction of research data from unstructured text—time-consuming and expensive. Semi-supervised methods like weakly supervised learning can circumvent the need for annotation. Part of the work of this dissertation has been on applying weak supervision to the automatic de-identification task. Furthermore, methods to make the annotation corpus denser with relevant information can increase the efficiency of the annotation process [45]. This has also been a focus of the work of this dissertation.

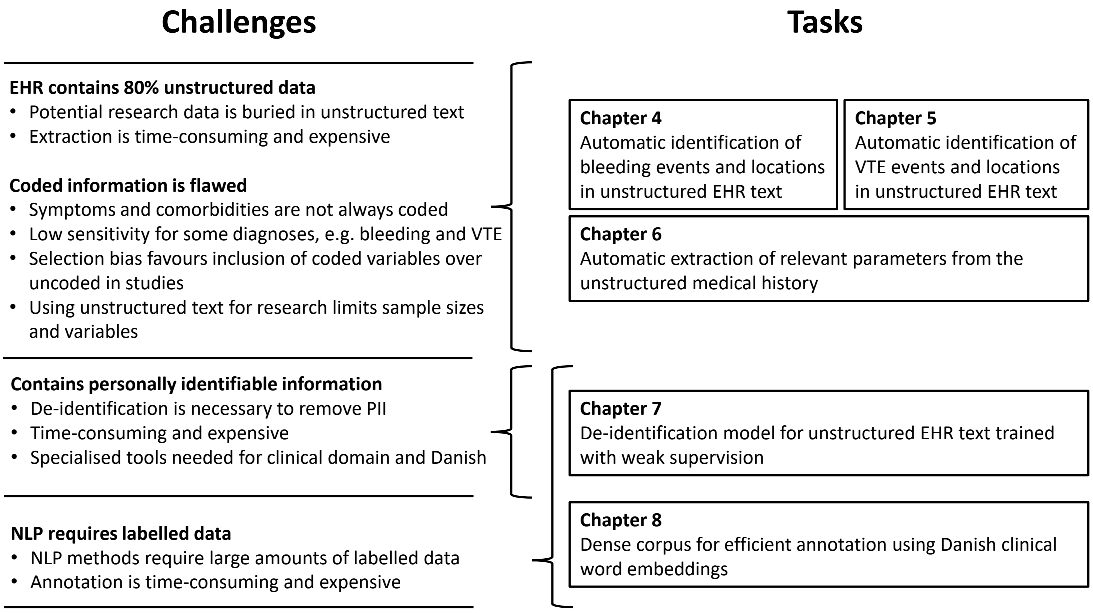

FIGURE 1.2: An overview of presented challenges and the tasks they prompt. Abbreviations: EHR, electronic health record; VTE, venous thromboembolism; PII, personal identifiable information.

Figure 1.2 shows an overview of the challenges presented in this introduction and the tasks they prompt. The tasks are formulated as research questions in the next section.

1.2 Problem Statement

Based on the challenges to clinical research outlined in the introduction, I define the following problem statement for this dissertation:

Problem Statement

The overall purpose of this PhD dissertation is to enable Danish clinical research by automatically extracting research data on bleeding and VTE from unstructured EHR text, and by extracting a structured overview of the medical history. Models to remove PII and make annotation more efficient will be developed to facilitate clinical research and model development.

The problem statement is addressed through five research questions, which are presented and briefly motivated below:

Reliable research data on bleeding and VTE are currently only found in unstructured EHR text. Automatic extraction of research data on bleeding and VTE from unstructured EHR text can improve the quality of studies and circumvent the time-consuming and expensive manual extraction. Therefore, I:

1. develop a deep learning pipeline to automatically extract research data on bleeding by identifying bleeding events and their anatomical location in Danish unstructured EHR text.

and

2. develop a deep learning pipeline to automatically extract clinical research data on VTE by identifying VTE events and their anatomical location in Danish unstructured EHR text.

The unstructured EHR text includes rich information on all relevant information from the medical history, which are not all coded in ICD codes and medical databases. All information from the medical history can be relevant for clinical research. Therefore, I:

3. develop a system for automatic extraction of relevant parameters from the medical history in the form of clinical events, their attributes, and relations contained in Danish unstructured EHR text.

Unstructured EHR text contain PII which means that explicit consent or de-identification is necessary before processing. Manual de-identification is infeasible for large sample sizes, and the development of automatic de-identification models specialised for Danish clinical text requires large amounts of labelled data for training. Therefore, I:

4. develop a deep learning model for automatic de-identification of names, streets, and locations in Danish unstructured EHR text using weak supervision.

NLP models require large amounts of labelled data to train, and the annotation process is time-consuming and expensive. Therefore, I:

5. train Danish clinical word embeddings that can be used to create a dense clinical annotation corpus and increase the efficiency of the annotation process.

1.3 Contributions

I published nine papers during my PhD studies. Five papers constitute the main contribution of the dissertation and are included in their entirety (in order of appearance):

- Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Rasmus Søgaard Hansen, Thiusius Rajeeth Savarimuthu, Rasmus Bank Lynggaard, and Pernille Just Vinholt. "Doctors identify hemorrhage better during chart review when assisted by artificial intelligence". In: *Applied Clinical Informatics*, 2023

- Jannik S Pedersen, Martin S Laursen, Cristina Soguero-Ruiz, Thiusius R Savarimuthu, Rasmus Søgaard Hansen, and Pernille J Vinholt. "Domain over size: Clinical ELECTRA surpasses general BERT for bleeding site classification in the free text of electronic health records". In: *2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE. 2022, pp. 1–4

- Martin Laursen, Jannik Pedersen, Rasmus Hansen, Thiusius Rajeeth Savarimuthu, and Pernille Vinholt. "Danish Clinical Named Entity Recognition and Relation Extraction". In: *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 655–666

- Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Vinholt, and Thiusius R Savarimuthu. "Automatic Annotation of Training Data for Deep Learning Based De-identification of Narrative Clinical Text". In: *WNLPe-Health 2022: Proceedings of The First Workshop on Context-aware NLP in eHealth (WNLPe-Health 2022)*. CEUR Workshop Proceedings. 2023, pp. 30–44

- Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Just Vinholt, Rasmus Søgaard Hansen, and Thiusius Rajeeth Savarimuthu. "Benchmark for evaluation of danish clinical word embeddings". In: *Northern European Journal of Language Technology* vol. 9, no. 1, 2023

I describe part of the work undertaken in the below paper. It is included as an appendix.

- Jannik Pedersen, Martin Laursen, Pernille Vinholt, and Thiusius Rajeeth Savarimuthu. "MeDa-BERT: A medical Danish pretrained transformer model". In: *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 301–307

The following papers were published but are not included in this dissertation:

- Jannik S Pedersen, Martin S Laursen, Thiusius Rajeeth Savarimuthu, Rasmus Søgaard Hansen, Anne Bryde Alnor, Kristian Voss Bjerre, Ina Mathilde Kjær, Charlotte Gils, Anne-Sofie Faarvang Thorsen, Eline Sandvig Andersen, Søren Andreas Just, and Pernille Just Vinholt. "Deep learning detects and visualizes bleeding events in electronic health records". In: *Research and practice in thrombosis and haemostasis* vol. 5, no. 4, 2021, e12505
- Martin S Laursen, Jannik S Pedersen, Søren A Just, Thiusius R Savarimuthu, Brian Blomholt, Jakob KH Andersen, and Pernille J Vinholt. "Factors Facilitating the Acceptance of Diagnostic Robots in Healthcare: A Survey". In: *2022 IEEE 10th International Conference on Healthcare Informatics (ICHI)*. IEEE. 2022, pp. 442–448
- Jannik Pedersen, Martin Laursen, Pernille Vinholt, Anne Alnor, and Thiusius Savarimuthu. "Investigating anatomical bias in clinical machine learning algorithms". In: *Findings of the Association for Computational Linguistics: EACL 2023*. 2023, pp. 1368–1380

1.4 Reading Guide

The rest of this dissertation is organised in the order of the research questions.

This dissertation is organised as an anthology. When an included paper or the work of an included paper is presented, its reference appears with all bibliographic details and copyright details. Some included papers contain parts that are not specifically relevant to this dissertation. Therefore, before the inclusion of a paper, its work is summarised in the context of the dissertation.

An understanding of deep learning and machine learning in general is assumed for technical descriptions. Except for the included papers, model architectures will be presented with limited details on hyperparameters.

Abbreviations are introduced at first occurrence and used throughout the dissertation. When needed, the pronoun "we" is used when describing the work of papers to reference all authors of the paper, and "I" is used in other cases.

The dissertation consists of the following chapters:

- **Chapter 2** is a brief introduction to NLP. Recurrent neural networks (RNNs), convolutional neural networks (CNNs), and the transformer architecture are described.
- **Chapter 3** outlines relevant related work for each of the deep-learning-based extraction and classification tasks described in the research questions.

- **Chapter 4** addresses research question 1. I first describe pre-trained large language models (LLMs) and proceed to present two papers where a bleeding event classification model and bleeding location classification model were developed. I end the chapter with concluding remarks on limitations of the models, comparing the pipeline's performance to that of ICD codes, a discussion of having a human in the loop to validate the models' predictions, and the application of the models to current and future clinical studies.
- **Chapter 5** addresses research question 2. I first describe the pre-trained LLM that was fine-tuned to develop a VTE event classification model and VTE location classification model. Then, the performances of the models are provided. Concluding remarks include limitations and future application to clinical studies.
- **Chapter 6** addresses research question 3. I first give an overview of named entity recognition (NER) and relation extraction (RE). Next, I present a paper that trains a NER and RE model for extraction of the medical history from Danish unstructured EHR text. Concluding remarks include a discussion of extraction versus classification.
- **Chapter 7** addresses research question 4. I first describe the need for de-identification of clinical data and what it includes. Next, I present weakly supervised learning as a means of reducing time spent on annotations. Finally, I present a paper that uses weak supervision to train a de-identification model for Danish unstructured EHR text.
- **Chapter 8** addresses research question 5. I describe a method to make the annotation process more efficient by creating a dense clinical annotation corpus with the use of clinical word embeddings. Then, I describe non-contextual word embeddings and present a paper that develops a Danish clinical word embedding benchmark and trains two Danish clinical word embedding models. Concluding remarks include a discussion of the effect of using a dense annotation corpus on the distribution of the negative class and how the method can be applied to future work to reduce time spent on annotation.

CHAPTER 2

A Brief Introduction to Natural Language Processing

NLP is the application of computational techniques to the analysis or synthesis of natural language text or speech[1]. NLP began in the 1950s and is an intersection of artificial intelligence and linguistics [55]. It has its roots in rule-based, symbolic approaches but is now dominated by machine learning methods, namely deep learning where deep neural networks are trained on labelled examples to be able to classify or generate unseen data [56].

In this chapter, I present the main deep learning architectures for text processing applied in related work and this dissertation. First, I give a brief overview of RNNs and CNNs before I present the transformer architecture which is the basis of state-of-the-art models in NLP [57, 58].

2.1 Recurrent Neural Networks

An RNN [59] is a deep learning architecture that is ideal for sequence modelling, i.e. processing of timeseries where the location of a datapoint in the sequence is important. Language can be considered a timeseries, and the RNN is a popular deep learning architecture for NLP [60]. The simple RNN processes words sequentially by generating a series of hidden states where the next hidden state, h_t, is a function of the previous hidden state, h_{t-1}, and the input at position t [61]. The hidden state can be considered keeping a history of the input sequence that is updated with every new input. After the last input, the hidden state is dependent on all words of the input sequence and can be classified. Formally,

$$h_t = \sigma(W_h h_{t-1} + W_x x_t)$$

[1] Definition from Oxford Languages

$$\hat{y} = softmax(W_z h_t)$$

where σ is the sigmoid activation function; h_t is the hidden state at timestep t; x_t is the input representation at timestep t; and W_h, W_x, and W_z are learned matrices [62].

Each input word representation, x_t, is a real-valued word embedding vector of dimensions d. All word embeddings in the vocabulary of size V are stored in the word embedding matrix $W_{d \times V}$. The word embedding matrix can be initialised from scratch and learned with the task [63] or used as pre-trained features. When used as pre-trained features, the word embedding matrix has been pre-trained on a general language task to encode language features in the embeddings. Methods include Continuous Bag-of-Words (CBOW) [64], Continuous Skip-gram [64], FastText [65], and Global Vectors (GloVe) [66].

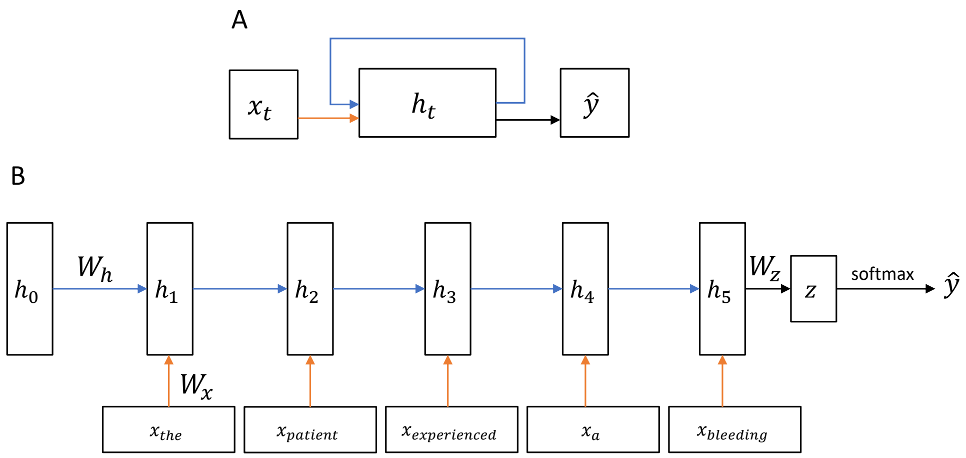

FIGURE 2.1: A: A recurrent neural network (RNN). B: An unfolded RNN showing the timesteps. Blue line: multiply by W_h. Orange line: multiply by W_x. h_t takes the sigmoid function of the added inputs.

Figure 2.1 shows a simple RNN for sequence classification. If an output classification is made for the hidden state of each timestep instead of only the final hidden state, a class can be predicted for each word in the input sequence. This is utilised in, e.g. NER [67]. Another utilisation is for sequence-to-sequence (seq2seq) transformation using an encoder–decoder structure, which is useful for, e.g. translation or summarisation [68].

RNNs can deal with position invariance and process dynamic, unlimited sequences. In practice, though, they do not learn long-term dependencies well because of vanishing gradients [69], and it is an issue to compress very long sequences into a single hidden state [70]. The long short-term memory cell [69] (LSTM) and the gated recurrent unit [71] (GRU) alleviate these problems by replacing the simple hidden layer calculation in the RNN by a more complex calculation involving gates that allow the RNN to learn what to keep and what to forget when processing the sequence.

2.2 Convolutional Neural Networks

CNNs [72] were first applied to computer vision and later to NLP [73]. CNNs feed the input embeddings into one or multiple convolutional layers. The convolutional layers each contain multiple kernels of a certain size that act as filters that learn during training to recognise patterns. Each of the filters in the convolutional layer learns to recognise different features from the same n-gram region of the input. The filters can take different sizes to recognise features in n-grams of different sizes [74].

The convolutional layers work by having the filters make one-dimensional convolutions on the input. The filters of the first convolutional layer act on the input embeddings while the filters of the following convolutional layers act on the output features of the previous layer. The deeper the architecture, the more abstract and complex features are extracted.

The input to the CNN is a matrix of dimensions $n \times d$ where each row is a word embedding, $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$, in the input sequence of n words, and d is the dimension of the word embeddings [75].

The embedding matrix consists of n-grams, u_i:

$$u_i = [x_i, \ldots, x_{i+l-1}] \in \mathbb{R}^{d \times l}; 0 \leq i \leq n - l$$

where n is the number of words in the input and l is the size of the filters (in words).

The convolutional layer consists of m filters:

$$f_j \in \mathbb{R}^{d \times l}$$

Each filter, f_j, slides over the input, making convolutions with each n-gram, u_i:

$$F_{ij} =< u_i, f_j >$$

The result is a feature matrix $F \in \mathbb{R}^{n \times m}$.

Max-pooling is then applied along the i'th dimension which results in $p \in \mathbb{R}^m$:

$$p_j = \underset{max}{i} F_{ij}$$

Each number in p represents the highest activation of that filter for the input. p is then passed through a rectified linear unit (ReLU) activation function, a fully-connected layer, and a softmax function to give the output distribution.

Figure 2.2 shows the process of applying multiple filters of different sizes to the input in parallel convolutional layers.

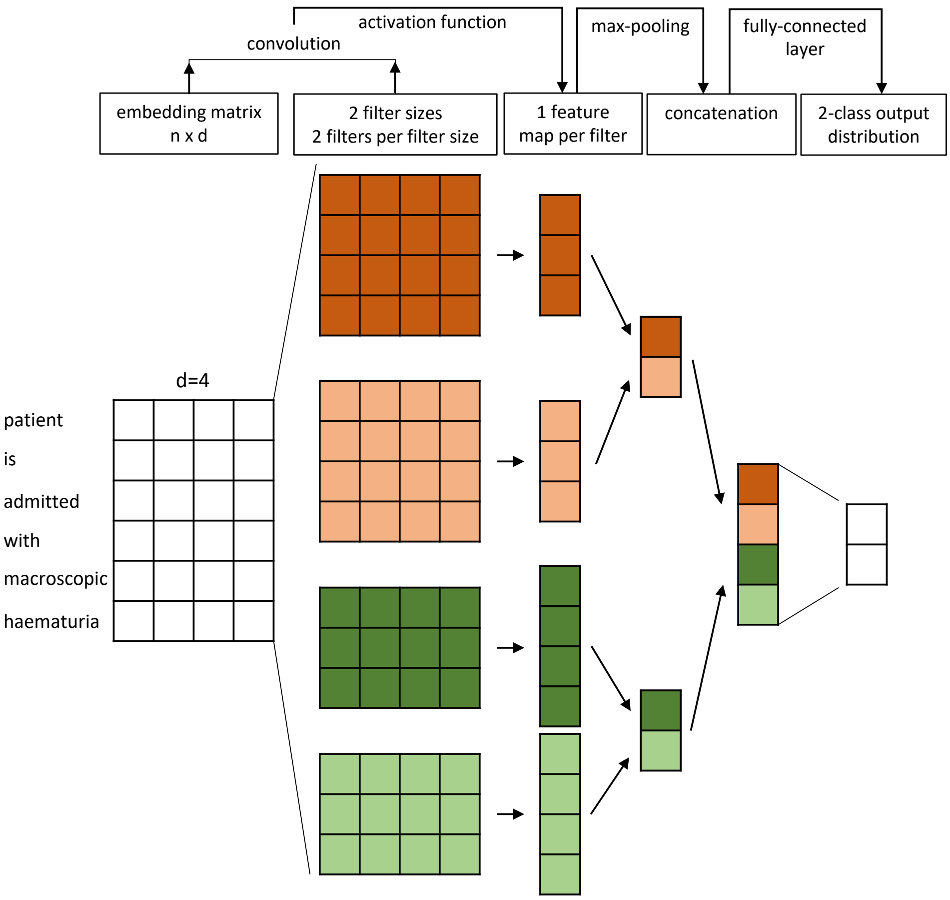

FIGURE 2.2: The process of applying multiple filters of different sizes to the input in parallel convolutional layers. Figure inspired by Zhang and Wallace [74]. Abbreviations: n, number of words in input; d, dimension of word embeddings.

2.3 Transformers

The transformer is a deep learning architecture that was presented in 2017 by Vaswani et al. in the paper "Attention is all you need" [61]. The original transformer architecture is a seq2seq model that uses an encoder–decoder structure for, e.g. machine translation. In seq2seq modelling, the encoder maps the input sequence, $x_1, \ldots, x_n$, into a sequence of continuous representations, $z = z_1, \ldots, z_n$. The decoder is auto-regressive and generates the output sequence, $y_1, \ldots, y_m$, one element at a time by decoding z and taking the previously generated output as additional input at each step of the decoding process [61]. With its introduction, the transformer replaced RNNs as the state-of-the-art in language modelling and machine translation [61].

While the transformer was developed for seq2seq modelling, the encoder and decoder parts have been used for two different kinds of LLMs. While the decoder has been used in unidirectional auto-regressive LLMs such as the generative pre-trained transformer (GPT) series including GPT-3, ChatGPT [76], and GPT-4 [77] to generate language, the encoder has been used for pre-trained bidirectional LLM encoders that can be fine-tuned to model specific classification tasks.

Below, I first give a broad overview of the entire transformer architecture. Then, I explain in further detail the input embeddings, and finally the encoder block as it is the basis of LLMs "Bidirectional Encoder Representations from Transformers" (BERT) [78] and "Efficiently Learning an Encoder that Classifies Token Replacements Accurately" (ELECTRA) [79], which are used in this dissertation. They will be presented in detail in chapter 4.

2.3.1 Architecture

Before the introduction of the transformer, both the encoder and decoder in seq2seq modelling would be represented by an RNN, and the decoder could attend to all hidden states of the encoder by applying an attention mechanism [70]. This alleviated the problems with long-term dependencies between early tokens and the decoder's output when encoding long sequences into a single hidden state.

In "Attention is all you need", the transformer architecture discards the recurrent structure in the encoder–decoder and instead relies only on an attention mechanism called self-attention when generating representations. Self-attention lets the encoder attend to other words in the input sequence when encoding a specific word, which makes the representations contextual—contrary to previous word representations. Instead of encoding the input sequence recurrently, the transformer encodes the whole sequence at once, in parallel. This decreases training time and makes it easier to encode long-range dependencies as the gradients do not have to flow through multiple hidden states [61].

Figure 2.3 shows the original transformer architecture. The transformer is composed of an encoder consisting of six encoder blocks and a decoder consisting of six decoder blocks. Each block feeds its output as input to the next block, making the representations increasingly complex.

The encoder takes the input sequence represented by word embeddings in parallel. The encoder block's main part is the multi-head self-attention mechanism which consists of multiple self-attention heads. Self-attention calculates how much information from each position in the input should be encoded in the representation of the current position, thereby making the word representations contextual. The output of the encoder stack is a contextual representation of each position of the input sequence.

The decoder block is architecturally like the encoder block except that it applies two forms of multi-head attention. First, it applies multi-head self-attention to its input, like the mechanism in the encoder block. Next, it applies a multi-head encoder–decoder attention that lets the decoder block calculate what positions of the encoder stack output are appropriate to attend to at this position in the decoding. Because the decoder is auto-regressive, using its outputs as

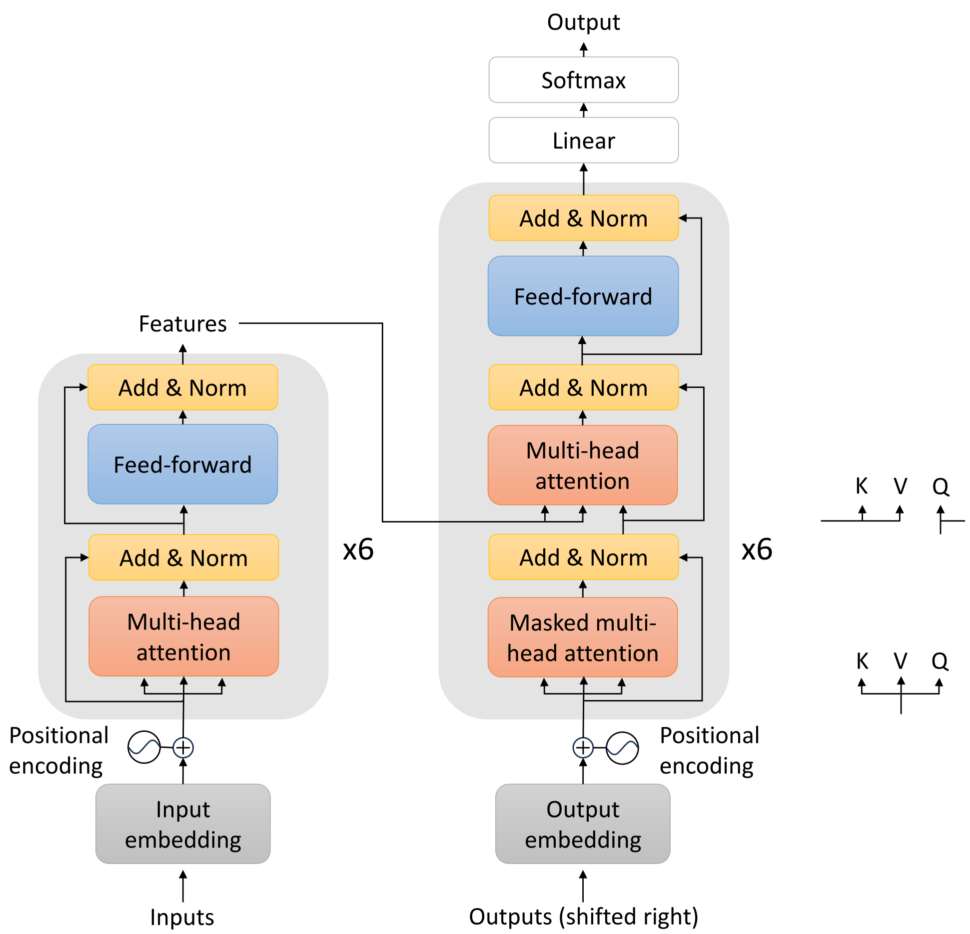

FIGURE 2.3: The transformer encoder–decoder architecture. K, V, and Q are the keys, values, and queries, which are different projections of the input. Inspired by Vaswani et al. [61].

inputs, the decoder's multi-head self-attention is only allowed to attend to earlier positions in the output. This is achieved by masking the other positions.

The decoding starts with a special start token as input to begin generating the output one token at a time. The output sequence is shifted right, making the input to the decoder all token predictions so far. The output of the decoder stack is passed through an output softmax layer over the vocabulary, and the predicted word is added to the decoder input sequence. This process continues until a special end token is generated.

2.3.2 Input Embeddings

While the five top encoder blocks take the output of the previous encoder block as input, the bottom encoder block takes a list of 512-dimensional (d_{model}) embeddings as input. These embeddings are learned during training. Unlike previous seq2seq models, the transformer does not process the input sequentially, but all at once. While this allows for parallelisation, information about the order of words is lost. For this reason, the input embeddings are summed with a positional encoding signal that encodes the position of the word in the sequence. The position encoding consists of two interweaved sinusoids.

$$sin(pos/10000^{2i/d_{model}})$$

is used to calculate the value for even indices of the 512-dimensional positional embedding, while

$$cos(pos/10000^{2i/d_{model}})$$

is used for the uneven indices. i is the index in the positional embedding and pos is the position of the word in the sequence. The resulting input embedding matrix carries information on both the meaning and position of the word in the sequence.

2.3.3 Encoder Block

The input sequence is passed to the bottom encoder block in the form of a list of embeddings enhanced with a positional encoding. Each subsequent encoder block passes its output as input to the next block. The encoder block consists of multi-head self-attention, a feed-forward neural network, and residual connections with normalisation.

The multi-head self-attention consists of $h = 8$ self-attention heads that let the encoder block attend to all other positions in the input sequence as it encodes the current position. This means that each position of the input sequence is represented by a weighted sum of all the input sequence's representations [80]. The output of one attention head is calculated as

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_{model}/h}})V$$

where

$$Q, K, V \in \mathbb{R}^{seq\_length \times (d_{model}/h)}$$

are the queries, keys, and values, which are projected from the input.

Simply, the query can be seen as capturing what information should be attended to for the current position. The key can be seen as indexing what kind of information each word in the sequence contains. The compatibility function is applied for the query and the key of each of the positions in the sequence—thereby calculating how much information from each position in the sequence should be encoded in the representation at the current position. After dividing the compatibility scores by $\sqrt{(d_{model}/h)}$ to avoid unstable gradients, the softmax function is taken over the whole sequence to obtain the weights on the values. The values are a simple transformation of the input [80]. The output at each position is a sum of the weighted value vectors.

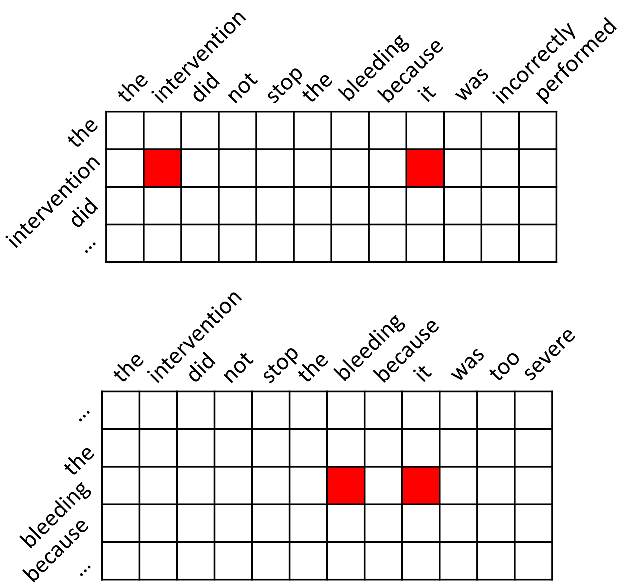

FIGURE 2.4: A simplified illustration of a self-attention map for two sentences. Red: high attention. Highlights are only made for "intervention" and "bleeding", respectively. Inspired by Elhariri [81].

An example with two sentences inspired by Alammar illustrates the function of self-attention [82]:

"**the intervention** did not stop the bleeding because **it** was incorrectly performed"

"the intervention did not stop **the bleeding** because **it** was too severe"

In the first sentence, "it" is referring to "the intervention" and should encode this in the representation while in the second sentence, "it" should encode information on "the bleeding". Figure 2.4 shows what could be an example of a resulting self-attention map for the two sentence examples. In the first sentence there is high attention on "it" for "intervention", and for the second sentence there is high attention on "it" for "bleeding". Both words also have a high attention on themselves [82].

Multi-head self-attention consists of applying self-attention $h = 8$ times in parallel. This allows the model to capture different representations of the words' relations [81]. According to the formulas presented in the paper, the input embedding matrix, $X \in \mathbb{R}^{seq\_length \times d_{model}}$, is projected to query, key, and value matrices of the same dimension:

$$Q = XW^Q$$

$$K = XW^K$$

$$V = XW^V$$

where

$$W^Q, W^K, W^V \in \mathbb{R}^{d_{model} \times d_{model}}$$

For multi-head self-attention, each of the $h = 8$ self-attention heads has its own learned parameters for projecting the queries, keys, and values to a subspace. Self-attention is then applied to those subspaces and the h outputs are concatenated and projected to the final multi-head self-attention output with the same dimension as the original input:

$$MultiHead(Q, K, V) \in \mathbb{R}^{seq\_length \times d_{model}} = Concat(head_1, \ldots, head_h)W^O$$

where

$$head_i \in \mathbb{R}^{seq\_length \times (d_{model}/h)} = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

and

$$W^O \in \mathbb{R}^{d_{model} \times d_{model}}$$

$$W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_{model} \times (d_{model}/h)}$$

In a residual connection around the multi-head self-attention, the input and output are added, and layer normalisation is applied. The residual connection allows gradients to flow through a shortcut in the architecture without any attenuation [83] while layer normalisation ensures faster training by preventing shifting means and standard deviations of the input features [84].

The output of the layer normalisation is a matrix of contextual representations where each row is a position in the input sequence. It is passed through a feed-forward neural network that is applied identically to each position with a residual connection and layer normalisation. The feed-forward neural network has a hidden layer of size 2048, a ReLU activation function, and an output layer of size $d_{model} = 512$. Figure 2.5 illustrates how the encoder block processes the input.

The output of the feed-forward neural network is then passed to the next encoder block with the same architecture. The output of the last of the six encoder blocks is the final contextual word representation with the same dimensions as the input embedding.

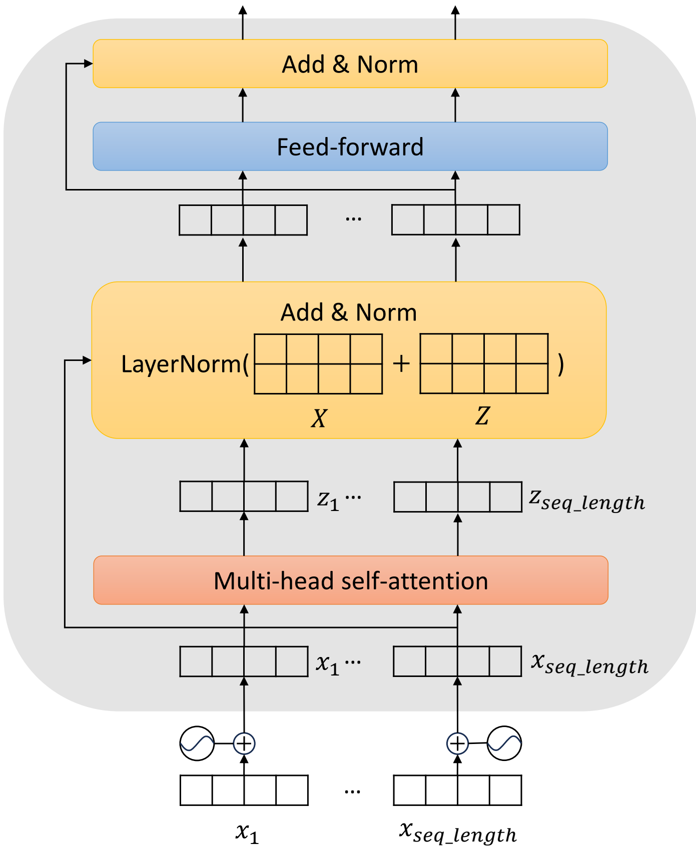

FIGURE 2.5: An overview of how the encoder block processes the input. $x_1, \ldots, x_{seq\_length}$ is the input to a multi-head self-attention layer and $z_1, \ldots, z_{seq\_length}$ is the output. In the residual connection, input and output are summed and layer normalisation is applied. That is repeated for the feed-forward neural network. Inspired by Alammar [82].

CHAPTER 3

Related Work

In this chapter, I present relevant related work for each of the deep-learning-based extraction and classification tasks described in the research questions. Notice that research question 5 falls outside of this category. The chapter is divided into sections corresponding to each research question.

First, I outline the related work on identification of bleeding and VTE events from unstructured EHR text. In the next two sections, I outline related work on extraction of multiple parameters of the medical history from unstructured EHR text, and de-identification of unstructured EHR text. I mainly focus on work published in or after 2017 where the transformer was invented unless a study presents work that is notably different to the rest.

I searched the following databases: ACL Anthology, which hosts papers on the study of computational linguistics and natural language processing; IEEE Xplore, a digital library containing nearly a third of the world's technical literature in electrical engineering, computer science, and electronics; PubMed, containing more than 35 million citations and abstracts of biomedical literature; and Google Scholar, for searching broadly in scholarly literature.

When searching ACL Anthology, I used variations of broad medical search terms such as "bleeding", "thrombosis", "health record", or "de-identification". For IEEE Xplore, I also used medical search terms with added variations of NLP-centred search terms such as "NLP", "text", or "extract". For PubMed, I used narrow medical or technical search terms such as "haemorrhage", "VTE", "NLP", or "named entity recognition". For Google Scholar, I used variations of broad and narrow search terms.

3.1 Identification of Bleeding Events

Related work on identification of bleeding events includes rule-based methods, traditional machine learning, traditional deep learning, and transformers.

Using rule-based methods, Taggart et al. classified bleeding events in clinical notes [85]. They developed a dictionary of terms indicating bleeding and their possible negations. The rules were built from a training set of 990 notes and tested on 660 notes. It performed with a sensitivity of 91.1%, specificity of 84.6%, and precision of 62.7%. Apart from the rule-based method, they also tested a support vector machine (SVM), a decision-tree-based method, and a CNN. All performed worse than the rule-based method.

Using traditional machine learning, Lee et al. detected patients that had bleeding caused by clopidogrel medication [86]. They extracted the features: drug names and signature information, clinical entities, bleeding ICD codes and identifiers, non-negated keywords related to bleeding, and clopidogrel information. Using the features as input, they trained a SVM on clinical notes of 1,512 patients and tested it on 756 patients. The SVM performed with sensitivity of 96.6%, precision of 88.7%, and F1 score of 92.48%. They also tested a rule-based classifier and scoring-based method that performed worse than the SVM.

Using traditional deep learning, Jnawali et al. classified intracranial bleeding in radiology reports [87]. They used a hybrid deep learning model consisting of a CNN to extract co-located features and an LSTM to extract sequential features. The data for training and testing the model were 12,851 computed tomography (CT) reports containing 1,886 positive cases and 10,966 negatives. The model performed with a sensitivity of 75%, specificity of 95%, and precision of 72%.

Li et al. classified bleeding events on sentence level in EHR notes [88]. They used a hybrid model consisting of a CNN and bidirectional LSTM (BiLSTM) autoencoder. The data used for training and testing were 878 EHR notes containing 76,577 sentences. The model performed with an F1 93.8%.

Mitra et al. classified bleeding events and their anatomical location in discharge summaries using a conditional random field (CRF) model and a LSTM-CRF model [89]. The data for training and testing were 1,079 EHR discharge summaries with six entity types including bleeding and bleeding location. The CRF model performed best for bleeding events with a sensitivity of 77%, precision of 83%, and F1 of 80%. The LSTM-CRF model performed best for anatomical location with a sensitivity of 75%, precision of 74%, and F1 of 74%. The authors also tested a biomedical BERT (BioBERT) transformer that performed worse than the traditional machine learning methods. They argued that it was due to the samples exceeding the sequence length limit of BERT, truncating potential important information for the classification.

3.2 Identification of Venous Thromboembolism Events

Related work on identification of VTE events includes rule-based and traditional machine learning methods.

Tian et al. classified DVT and PE in narrative radiology reports [90]. For each of DVT and PE, they developed a rule-based classifier using unigrams, bigrams, and negations, utilising 4,000 narrative reports from VTE imaging studies. The DVT model performed with a sensitivity of 94%, specificity of 96%, and precision of 73%. The PE model performed with a sensitivity of 94%, specificity og 96%, and precision of 80%.

Gálvez et al. classified VTE in radiology reports using a rule-based method based on Systemized Nomenclature of Medicine terms associated with thrombosis and other selected relevant terms [91]. They used 250 ultrasound reports for development and 6,373 radiology reports for testing. The method had a sensitivity of 82.9%, specificity of 97.5%, and F1 score of 66%.

Dantes et al. classified VTE in EHRs with a rule-based method using selected terminology and contextual information, such as report sections [92]. The method was evaluated on 2,083 radiology reports and 422 surgical encounters with radiographic studies for VTE. For radiology reports, the method performed with a sensitivity of 97.2% and specificity of 99.3%. For surgical encounters, the method performed with a sensitivity of 97.6% and specificity of 99.8%.

Shah et al. classified the presence of VTE along with other complications in unstructured text of medical reports on patients undergoing primary arthroplasty [93]. Their method utilised tokenization, normalisation, dictionary look-up, negation detection, and keyword searches. On 1,000 randomly selected clinical and hospital notes for patients undergoing primary arthroplasty, the method performed with an accuracy of 90%, sensitivity of 97%, and specificity of 86%.

Johnson et al. classified PE and the affected pulmonary artery in radiology reports [94]. Their rule-based method was based on a dictionary containing terms related to anatomical location, laterality, temporality, and certainty of the PE. 280 radiology reports were used for development of the method and 120 for testing. The method performed with a sensitivity of 96.0%, specificity of 97.7%, and precision of 86.3% for identifying PE. For classification of PE location, sensitivity ranged between 7.0% and 93.3%, and specificity between 96.5% and 100%.

Verma et al. classified VTE in radiology reports with a rule-based method that was developed iteratively on 1,551 reports and tested on 4,498 [95]. Reports were from CT thorax scans for PE, nuclear ventilation/perfusion scans for PE, and extremity compression ultrasound scans for DVT. For CT reports, the model performed with a sensitivity of 91%, specificity of 99%, and precision of 89%. For ventilation/perfusion reports, the model performed with a sensitivity, specificity, and precision of 100%. For ultrasound reports, the model performed with a sensitivity of 94%, specificity of 99%, and precision of 90%.

Li et al. classified VTE in a longitudinal cohort of cancer patients [96]. Their method consisted of using a rule-based pipeline that included a VTE entity recogniser, assertion classifier, part-of-speech tagger, sentence detector, and tokenizer. The method was developed on 1,300 radiology reports and evaluated on 800 patients. It performed with a weighted sensitivity and precision of 90% and 80%, respectively.

Jin et al. classified VTE in clinical patient cases using a rule-based method consisting of manually designed rules and matching of disease definitions from guidelines, the Unified Medical Language System, and medical websites [97]. The method was developed on 200 clinical cases and tested on 260. The method performed with a sensitivity of 89.9%, specificity of 99.8%, and F1 of 90.0%.

Using tradition machine learning, Rochefort et al. classified DVT and PE in unstructured radiology reports using two separate SVM models [98]. The SVM models were trained with a bag-of-words approach using a document-by-term matrix. The data consisted of 2,000 radiology reports from patients with expected VTE. Using feature selection, the initial 62,416 distinct unigram and bigram features were reduced to 118 unigrams and 218 bigrams associated with DVT, and 301 unigrams and 1,242 bigrams associated with PE. The DVT model performed with a sensitivity of 80%, specificity of 98%, and precision of 89%. The PE model performed with a sensitivity of 79%, specificity of 99%, and precision of 84%.

3.3 Extraction of Medical History Parameters

Related work on extraction of relevant parameters from the medical history from unstructured EHR text includes traditional deep learning methods and the transformer architecture.

Using traditional deep learning, Lerner et al. performed clinical NER in French and English with a hybrid model [99]. The French dataset consisted of 147 notes and contained the entities: drug names, signs or symptoms, diseases or disorders, diagnostic procedures or lab tests, and therapeutic procedures. The English dataset was 268 discharge summaries with drug name labels. The authors used a training–validation–test split of 70%–15%–15%. The hybrid model consisted of a terminology system which used regular expressions to extract drug names and a supervised system consisting of a bidirectional GRU. An embedding that represented the class predicted by the terminology system was concatenated to the input word embedding processed by the supervised system. The hybrid model performed with an F1 of 92.2% on English and 86.4% on French.

Ge et al. used privacy-preserving federated learning to leverage data on different platforms to train a medical NER model [100]. Three NER datasets on three different platforms were used for the federated learning. There were 341 overlapping entities within the types: drug, adverse drug effect, disease, finding, symptom, and dosage, but not all datasets had all entity types. The authors used a 80%–20% training–test ratio. The model consisted of a CNN for processing pre-trained word embeddings, character embeddings, and language model embeddings followed by a BiLSTM and CRF layer. The CNN and embeddings were trained on all data sources and the rest of the architecture on each dataset. The three models trained with different local data achieved F1 scores of 65.16%, 82.57%, and 32.69%.

Using a transformer architecture, Yu et al. automatically annotated clinical problems, treatments, and tests in the EHR [101]. They used a BioBERT model which was initialised from a pre-trained BERT model and adaptively pre-trained on a corpus of PubMed abstracts and PubMed Central articles. The BioBERT was used to make token representations which were fed to a BiLSTM and then a CRF layer. 170 discharge summaries were used for training and 170 for testing. The model performed with a sensitivity of 87.13%, precision of 87.07%, and F1 of 87.10%.

Using both traditional deep learning and the transformer architecture, Christopoulou et al. extracted adverse drug effects and medication relations in EHRs [102]. They extracted the relations between drugs and: strength, dosage, duration, frequency, form, route, reason, and adverse drug effect. The model was an ensemble of deep learning methods consisting of a BiLSTM for intra-sentence RE and a transformer for inter-sentence RE. 44,475 sentences were used for training and 11,520 for development. The method performed with an F1 of 94.72% and 87.65% on two different test sets.

Using a transformer architecture, Schneider et al. performed clinical NER on two different datasets [103]. The 13 entities were: disorders, chemicals and drugs, medical procedure, diagnostic procedure, disease or syndrome, findings, health care activity, laboratory or test result, medical device, pharmacologic substance, quantitative concept, sign or symptom, and therapeutic or preventive procedure. The authors used a Portuguese BioBERT model which was initialised from a pre-trained multilingual BERT model and adaptively pre-trained on 2.1 million clinical notes from Brazilian hospitals. The authors used two datasets of 1,000 clinical notes and 281 neurology clinical case descriptions and applied a 60%–20%–20% training–validation–test split. For the first dataset, the model performed with a sensitivity of 60.7%, precision of 60.8%, and F1 of 60.4%. For the second dataset, the model performed with a sensitivity of 93.5%, precision of 91.7%, and F1 of 92.6%.

3.4 De-identification of Unstructured Electronic Health Record Text

Related work on de-identification of unstructured EHR text includes rule-based methods, traditional machine learning, traditional deep learning, and the transformer architecture.

Using a rule-based method, Pantazos et al. de-identified names, street names, zip codes, hospital and clinic names, personal identification numbers, phone numbers, e-mail addresses, and URLs in Danish EHRs while preserving medical correctness, readability, and consistency [104]. The rule-based method utilised dictionaries of identifiers from official sources and fields in the EHR as well as manually created rules. De-identification consisted of mapping existing identifiers to new ones; distorting personal identification numbers; and randomising phone numbers, e-mail addresses, and URLs. Identifiers that were found to be ambiguous, e.g. both a medical term and common word, were kept in the database if they appeared more than 200 times and otherwise the EHR was deleted. The method was tested on 369 full EHRs and performed with a sensitivity of 99.5%, precision of 92.3%, and F1 of 95.7%.

Using tradition machine learning, Berg and Dalianis de-identified age, full date, date part, first name, last name, health care unit, location, and phone number in Swedish unstructured EHRs, utilising a semi-supervised method [105]. The authors used three datasets: a labelled and unlabelled training set, and a labelled evaluation set. 62 labelled EHRs were used for training and 36 for evaluation. The unlabelled data were 20 times larger than the labelled. They used a CRF model that took tokens and lexical features as input. The method also utilised information on capitalisation and punctuation, and further used regular expressions to identify dates, phone numbers, and words occurring in lists of names, hospitals, and locations. The CRF was initially trained on the labelled dataset and then self-training was applied to provide pseudo labels to the unlabelled data and add those with high confidence to the training set. The method performed with a sensitivity of 89.20%, precision of 94.20%, and F1 of 91.63%.

Using traditional deep learning, Liu et al. de-identified name, profession, location, age, data, contact, ID, and their subcategories in the 2014 i2b2 and 2016 N-GRID de-identification datasets consisting of EHRs [106]. The 2014 i2b2 corpus has a training set of 790 records and a test set of 514 records. The 2016 N-GRID corpus has a training set of 600 records and a test set of 400 records. The two datasets contain all PII identifier types according to the United States Health Insurance Portability and Accountability Act (HIPAA) privacy rule [107] and the two datasets are annotated using the same guidelines. The authors developed a hybrid system consisting of an ensemble of four subsystems: a BiLSTM for processing tokens; a separate BiLSTM and CRF model for processing other features such as EHR section information, dictionary features, and part-of-speech tags; and a rule-based system for extracting specific PII such as phone numbers. For the 2014 i2b2 dataset, the ensemble model performed with a sensitivity of 93.80%, precision of 96.46%, and F1 of 95.11%. For the 2016 N-GRID dataset, the ensemble model performed with a sensitivity of 88.81%, precision of 94.22%, and F1 of 91.43%.

Catelli et al. de-identified the PII categories as defined in the 2014 i2b2 de-identification dataset in an Italian COVID-19 dataset [108]. The dataset contained 115 COVID-19 EHRs of which 65 were used for training and development, and 50 for testing. The authors used a BiLSTM-CRF model which performed with an F1 of 92.94% on the seven main categories and 80.63% on subcategories. The authors also tested a BERT model which performed worse than the BiLSTM-CRF.

Using traditional deep learning and a transformer architecture, Tang et al. de-identified the 2014 i2b2 and 2016 N-GRID de-identification datasets [109]. They used a BiLSTM-CRF model with input representations generated by a pre-trained BERT. On the 2014 i2b2 dataset, it performed with a sensitivity of 95.02%, precision of 95.99%, and F1 of 95.50%. On the 2016 N-GRID dataset, it performed with a sensitivity of 90.31%, precision of 93.39%, and F1 of 91.82%.

Using a transformer architecture, Catelli et al. de-identified the 2014 i2b2 de-identification dataset [110]. The authors used an ELECTRA model which was enhanced by grouping sentences together before they were input to the model and thereby broadening the context of representations. It performed with an F1 of 96.37% on the main categories and 95.13% on the subcategories.

Liu et al. also de-identified the 2014 i2b2 de-identification dataset using a transformer architecture [111]. They used zero-shot learning by prompting GPT-4 to automatically identify and remove identifying information. The prompts were explicit, containing concrete information on the desired output and task, and provided examples. On 50 random samples from the 2014 i2b2 de-identification test set, the method achieved an accuracy of 99%, clearly surpassing other tested methods that fine-tuned LLMs like BERT.

CHAPTER 4

Identification of Bleeding Events

In this chapter, I present a deep learning pipeline to automatically extract research data on bleeding by identifying bleeding events and their anatomical location in Danish unstructured EHR text. This addresses research question 1.

As detailed in the introduction, bleeding is an important parameter for clinical research, and studies would benefit from automatic extraction of research data on bleeding events from unstructured EHR text. The anatomical location of a bleeding is an additional important parameter to extract as it partly defines if a bleeding is considered major. The anatomical location of a bleeding is relevant in, e.g. the International Medical Prevention Registry on Venous Thromboembolism (IMPROVE) bleeding risk study where only major or non-major but clinically relevant bleedings were included and found to increase bleeding risk [18]. The IMPROVE study defines major bleeding, in part, as bleeding occurring in a critical organ: intracranial bleeding, retroperitoneal bleeding, intraocular bleeding, adrenal gland bleeding, spinal bleeding, and pericardial bleeding. Non-major but clinically relevant bleeding is defined, in part, by bleeding in the following anatomical locations: overt GI bleeding, gross haematuria, substantial epistaxis, extensive haematoma or bruising, intraarticular bleeding, and menorrhagia or metrorrhagia [18].

Bleeding events can be difficult to extract from the unstructured EHR text. The text includes many rare words, abbreviations, a mix of languages, and spelling mistakes, which makes it difficult to extract with simple search terms [42]. Adding to the challenge, making simple searches in the unstructured text is not possible in all EHR systems. The fact that bleeding can be described in multitudes of different ways would lead to many false negatives for a simple search, and the inability to capture negations would lead to false positives. While the EHR systems employ different templates for different clinical applications, it is not guaranteed that the correct template is filled out, and the templates vary between EHR systems. This emphasises the need to rely on information in the unstructured text, which is mostly independent of the EHR system and template in use, when extracting.

In the rest of this chapter, I present a deep learning pipeline to automatically extract research data on bleeding events from unstructured EHR text. First, I introduce the concept of pre-trained LLMs and present the BERT and ELECTRA LLMs, which are used for the work presented in this chapter. Next, I present the two deep learning classification models that constitute the pipeline. The first model classifies bleeding events. The second model takes bleeding events as input and classifies their location in EHR text.

4.1 Pre-trained Large Language Models

The deep learning models for identifying bleeding events and their location from unstructured EHR text are based on the BERT and ELECTRA pre-trained LLMs. Both BERT and ELECTRA are based on the transformer encoder which produces contextual word representations [61]. The transformer encoder was presented in detail in section 2.3.3.

LLMs are pre-trained in a self-supervised fashion on a large unlabelled corpus such as Wikipedia, to model language. After pre-training, the LLMs can be used for transfer learning by fine-tuning the whole architecture for a specific downstream task, only by adding a task-specific output-layer. It has shown to improve performance on NLP tasks when fine-tuning a pre-trained LLM rather than training from scratch [78]. This allows for fewer labelled samples compared to training a task-specific architecture. Part of the reason is that a pre-trained LLM can relate words in an input sequence and few parameters need to be learned from scratch [78]. Task-specific architectures that take pre-trained non-contextual word embeddings as input features need to train to relate the words in the sequence.

Pre-trained LLMs are especially useful for low-resource languages like Danish [43] with a scarcity of labelled data, where a time-consuming and expensive annotation process must be undertaken to be able to train downstream tasks. Many pre-trained general LLMs are available online[1] but the clinical domain is underrepresented. Further, the clinical domain often requires specialised tools due to the specific terms and sentence structures used [42].

4.1.1 Bidirectional Encoder Representations from Transformers

"Bidirectional Encoder Representations from Transformers", or BERT [78], is a LLM with an architecture based on the transformer encoder, which produces contextual representations of each token. The original BERT model uses 12 encoder blocks, 12 self-attention heads per multi-head self-attention, and an embedding size of 768.

BERT uses a 30,000 token WordPiece [112] vocabulary. WordPiece is a tokenization method that uses sub-words to avoid out-of-vocabulary words and reduce the total size of the vocabulary. The method works by selecting, in the case of BERT, 30,000 wordpieces for the vocabulary

[1] E.g. `https://huggingface.co/`

that results in the minimum amount of wordpieces in the training corpus when tokenized according to the chosen vocabulary. The vocabulary can consist of both words and sub-words. Sub-words are prepended with "##" to be able to reconstruct the original sequence of words.

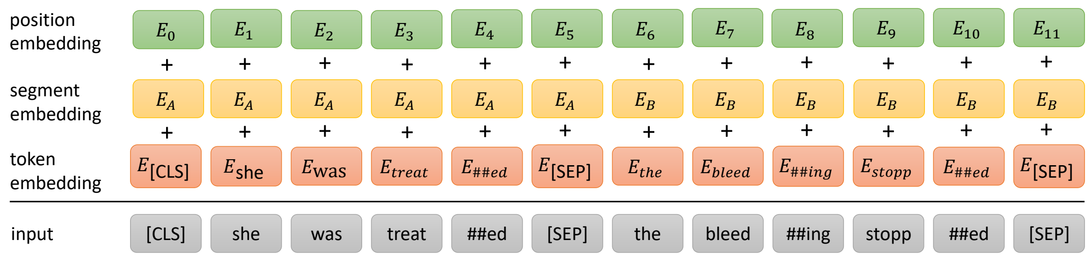

FIGURE 4.1: Illustration of the sentence pair "she was treated" and "the bleeding stopped" being input to BERT. The input representation is constructed by summing the token embeddings, segment embeddings, and position embeddings. Inspired by Devlin et al. [78].

BERT can take a sentence or a sentence pair, packed into one sequence, as input. The first sentence is prepended with a [CLS] token and followed by a [SEP] token. This corresponds to segment A. In case of a sentence pair, the second sentence follows, and finally a [SEP] token. This corresponds to segment B. Each input representation consists of the sum of a learned token embedding, segment embedding, and positional embedding. Since the position embedding is trained for a specific input length, BERT has a maximum input length before truncation will be applied. Figure 4.1 shows how the input representation is constructed by summing the token embeddings, segment embeddings, and position embeddings.

BERT is pre-trained using two unsupervised tasks: masked language modelling (MLM) and next sentence prediction (NSP).

In MLM, some tokens in the input sequence are replaced by a mask, and the original tokens are predicted by feeding the final contextual representation to an output softmax layer over the vocabulary. 15% of tokens in the input sequence are selected at random to be predicted. Only some are replaced with a [MASK] token to not create a mismatch during pre-training and fine-tuning where the [MASK] token does not occur. Of the 15%, 80% are replaced with a [MASK] token, 10% are replaced with a random token, and 10% remain unchanged.

NSP is applied to train BERT to, apart from word relationships, model sentence relationships. In NSP, a pair of sentences, A and B, are input to BERT. 50% of the time, B is the sentence that follows A in the training corpus. In the other 50%, B is a random sentence from the corpus. The [CLS] token is fed to an output softmax layer to classify either "IsNext" or "NotNext". Figure 4.2 illustrates the pre-training of BERT.

After BERT has been pre-trained for language representation, it can be fine-tuned for a specific downstream task using labelled data. In the case of sentence-level classification, the [CLS] token is fed to the output classification layer, and in the case of token-level classification, the token representations are used.

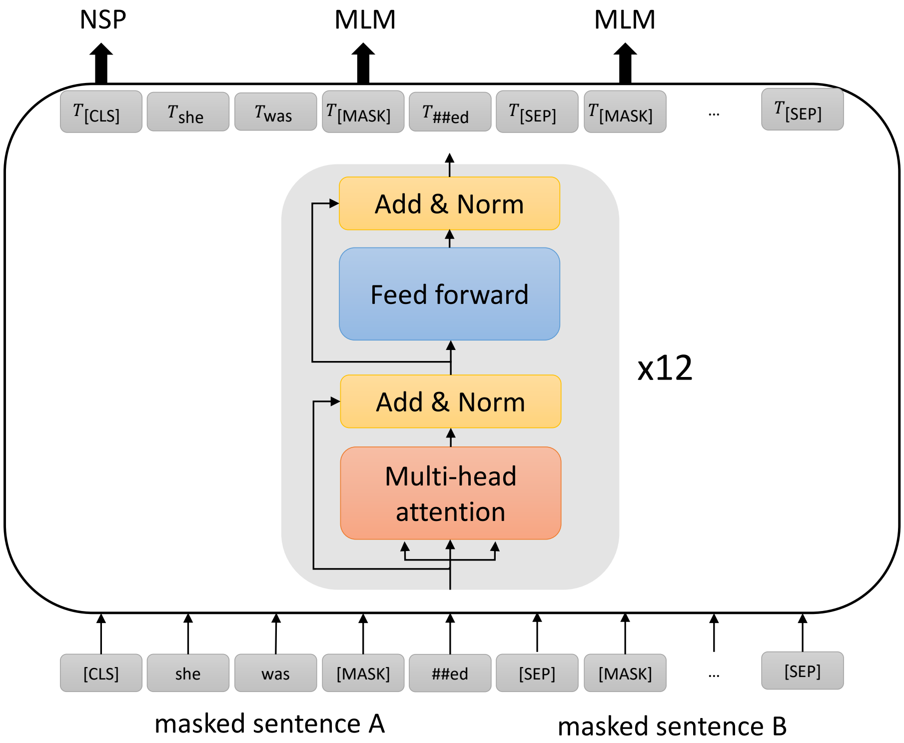

FIGURE 4.2: An overview of BERT pre-training on a sentence pair. Masked sentence A is: "she was treated". T is the final representation of the input token. BERT consists of 12 transformer encoder blocks. Inspired by Devlin et al. [78]. Abbreviations: NSP, next sentence prediction; MSM, masked language modelling.

4.1.2 Efficiently Learning an Encoder that Classifies Token Replacements Accurately

"Efficiently Learning an Encoder that Classifies Token Replacements Accurately", or ELECTRA [79], was proposed as an alternative to BERT. It implements a more efficient pre-training task called replaced token detection (RTD), replacing the MLM pre-training task of BERT, and discarding the NSP pre-training task.

In RTD, instead of masking some words in the input sequence as in MLM, they are replaced by a plausible alternative, and the model learns to distinguish real input tokens from replaced ones. An advantage is that the model learns from all input tokens instead of just the masked tokens. This makes the method more efficient with regards to compute and parameters, and it needs less training data than BERT to produce good results. This makes ELECTRA ideal for pre-training on small corpora [79].

ELECTRA consists of a generator and discriminator that are both a transformer encoder with the generator $1/4$–$1/2$ the size of the discriminator. The tokenization method is similar to that of BERT [113]. ELECTRA only needs one sentence for pre-training.

The generator is, like BERT, trained to perform MLM. A random set of input positions are replaced by a [MASK] token. The generator learns to predict the original token by passing the [MASK] token representation of the transformer encoder to an output softmax layer over the vocabulary.

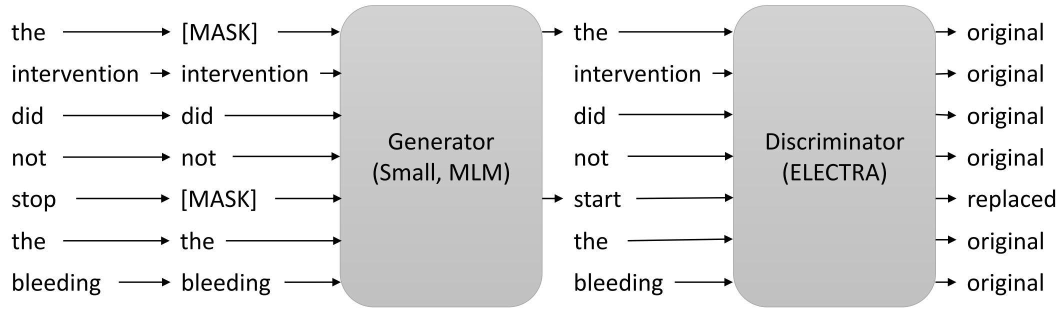

FIGURE 4.3: An overview of the replaced token detection pre-training method of ELECTRA. Inspired by Clark et al. [79]. Abbreviations: MLM, masked language modelling.

The discriminator takes the same input as the generator, but with the [MASK] tokens replaced by the generator's predictions. The discriminator predicts if each token is original or replaced by passing each encoder token representation through a 2-class sigmoid output layer. If the generator predicted the original token from the [MASK], the discriminator considers it original. Figure 4.3 shows an overview of the ELECTRA pre-training method.

During pre-training, the combined loss of the generator and discriminator is minimised over a large corpus. After pre-training, the generator is discarded and the discriminator is fine-tuned for downstream tasks, passing the [CLS] token to an output layer for sentence-level classification or the token representations for token-level classification.

4.2 Classification of Bleeding Events

This section presents a model to classify if sentences from unstructured EHR text indicate a bleeding event. It takes sentences as input and outputs the prediction. The model is the first part of the pipeline for identification of bleeding events and their location in unstructured EHR text. The work on the model is detailed in the following paper:

[46] Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Rasmus Søgaard Hansen, Thiusius Rajeeth Savarimuthu, Rasmus Bank Lynggaard, and Pernille Just Vinholt. "Doctors identify hemorrhage better during chart review when assisted by artificial intelligence". In: *Applied Clinical Informatics*, 2023

Below, I briefly present the contents of the paper in the context of this dissertation. The paper is included at the end of this section. Permission to reproduce this material was granted by Georg Thieme Verlag KG, © Georg Thieme Verlag KG.

In this paper, we fine-tuned of our own clinical ELECTRA LLM pre-trained on 1.4 billion words from 299,718 Danish EHRs from Odense University Hospital [47] (henceforth, referred to as Clin-ELECTRA) for the task of bleeding event classification. We used Clin-ELECTRA rather than a publicly available general domain pre-trained LLM to better represent the clinical language used in EHRs. We fine-tuned using 25,862 labelled sentences from 900 EHRs. The sentences had a binary label as positive or negative for indication of a bleeding event. A bleeding event could be any current, prior, or possible bleeding to cover all bleeding mentions that could be of interest to the healthcare staff.

An additional label indicated the location of the bleeding event as: central nervous system (CNS), eye, ear-nose-throat (ENT), airway, GI, internal, gynaecological, urological, muscle and joint, dermatological, multiple locations, or unknown location. The additional labels were used to make an even distribution of bleeding locations in the positive samples of the validation and test sets, allowing an evaluation of bias towards anatomical locations. Muscle and joint, and multiple locations were only in the training set as they had too few samples to balance the validation and test sets. We have documented the importance of this evaluation in other work [54]. Notice that, in in this paper, the location labels are only used for evaluation of anatomical bias. Classification of bleeding location is discussed in section 4.3. Additionally to developing and evaluating the bleeding event classification model, the paper also includes user studies that evaluate MDs' performance when extracting information from unstructured text with and without assistance from the model.

The paper contributes to this dissertation with a model that classifies if sentences indicate a bleeding event. The model is a fine-tuned Clin-ELECTRA LLM with a sensitivity and specificity of 95.8% on a test set balanced on bleeding indication and location. It has a sensitivity of 93.7% and specificity of 98.1% on an unbalanced test cohort of 566 admission. The model performs with sensitivities between 87.6% (dermatological) and 100% (muscle and joint, and eye) on different anatomical locations, and there are no major differences in sensitivity between minor and major bleeding, males and females, and young and elderly.

Doctors identify hemorrhage better during chart review when assisted by artificial intelligence

Authors

- Martin Sundahl Laursen*,[1], MSc, The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, msla@mmmi.sdu.dk
- Jannik Skyttegaard Pedersen*, MSc, The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, jasp@mmmi.sdu.dk
- Rasmus Søgaard Hansen, M.D., Department of Clinical Biochemistry, Odense University Hospital, rasmus.sogaard.hansen@rsyd.dk
- Thiusius Rajeeth Savarimuthu, Professor, The Maersk Mc-Kinney Moller Institute, trs@mmmi.sdu.dk
- Rasmus Bank Lynggaard, MSc, Department of Clinical Biochemistry, Odense University Hospital, rasmus.bank.lynggaard@rsyd.dk
- Pernille Just Vinholt, Chief M.D., Department of Clinical Biochemistry, Odense University Hospital, pernille.vinholt@rsyd.dk

*Equal contribution

[1] Corresponding author, Campusvej 55, 5230 Odense M, Denmark, +4565504469

Abstract

Objectives: This study evaluated if medical doctors could identify more hemorrhage events during chart review in a clinical setting when assisted by an artificial intelligence (AI) model, and medical doctors' perception of using the AI model.

Methods: To develop the AI model, sentences from 900 electronic health records were labeled as positive or negative for hemorrhage and categorized into one of twelve anatomical locations. The AI model was evaluated on a test cohort consisting of 566 admissions. Using eye-tracking technology, we investigated medical doctors' reading workflow during manual chart review. Moreover, we performed a clinical use study where medical doctors read two admissions with and without AI assistance to evaluate performance when, and perception of using the AI model.

Results: The AI model had a sensitivity of 93.7% and a specificity of 98.1% on the test cohort. In the use studies, we found that medical doctors missed more than 33% of relevant sentences when doing chart review without AI assistance. Hemorrhage events described in paragraphs were more often overlooked compared to bullet-pointed hemorrhage mentions. With AI assisted chart review, medical doctors identified 48 and 49 percentage points more hemorrhage events than without assistance in two admissions, and they were generally positive towards using the AI model as a supporting tool.

Conclusions: Medical doctors identified more hemorrhage events with AI assisted chart review and they were generally positive towards using the AI model.

Keywords: Electronic Health Records, Hemorrhage, Artificial Intelligence, Decision Support Systems, Natural Language Processing

4.3 Classification of Bleeding Location

This section presents a model that takes sentences from unstructured EHR text that indicate a bleeding event as input and classifies the anatomical location of the bleeding. The model is the second part of the pipeline for identification of bleeding events and their location in unstructured EHR text. The work on the model is detailed in the following paper:

[47] Jannik S Pedersen, Martin S Laursen, Cristina Soguero-Ruiz, Thiusius R Savarimuthu, Rasmus Søgaard Hansen, and Pernille J Vinholt. "Domain over size: Clinical ELECTRA surpasses general BERT for bleeding site classification in the free text of electronic health records". In: *2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE. 2022, pp. 1–4

Below, I briefly present the contents of the paper in the context of this dissertation. The paper is included at the end of this section. Reprinted with permission, from: Jannik Skyttegaard Pedersen, Martin Sundahl Laursen, Cristina Soguero-Ruiz, Thiusius Rajeeth Savarimuthu, Rasmus Søgaard Hansen, and Pernille Just Vinholt; Domain over size: Clinical ELECTRA surpasses general BERT for bleeding site classification in the free text of electronic health records; 2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI); September 2022; © 2022 IEEE.

In this paper, we fine-tuned the Clin-ELECTRA LLM on 17,344 sentences for the task of classifying the location of bleeding events. The model classifies locations: CNS, eye, ENT, airway, GI, internal, gynaecological, urogenital, dermatological, and unknown location. Notice that CNS bleedings are referred to as "cerebral" in the paper while dermatological bleedings are referred to as "skin". Muscle and joint bleedings are not classified because too few samples were available for multi-class training. Furthermore, the model does not classify multiple locations.

Location accuracy	
Location	**Accuracy, %**
Cerebral	96.67
Eyes	96.67
Ear-nose-throat	93.33
Airway	90.00
Gastrointestinal	94.00
Internal bleeding	78.67
Gynaecological	96.00
Urogenital	99.33
Skin	90.67
Unknown	79.33

TABLE 4.1: The fine-tuned Clin-ELECTRA model's accuracies by location.

The paper contributes to this dissertation with a model that classifies the location of the bleeding from sentences indicating a bleeding event. The model is a fine-tuned Clin-ELECTRA LLM with an overall accuracy of 90.5% on a balanced test set. Internal bleeding has the worst accuracy at 78.67% while urogenital bleeding has the best at 99.33%. I include the accuracies by location in table 4.1 as they are only shown in a bar plot in the paper.

Domain over size: Clinical ELECTRA surpasses general BERT for bleeding site classification in the free text of electronic health records

1st Jannik S. Pedersen*
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
Odense, Denmark
jasp@mmmi.sdu.dk

2nd Martin S. Laursen*
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
Odense, Denmark
msla@mmmi.sdu.dk

3rd Cristina Soguero-Ruiz
Signal Theory and Communication
Universidad Rey Juan Carlos,
Fuenlabrada, Spain
cristina.soguero@urjc.es

4th Thiusius R. Savarimuthu
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
Odense, Denmark
trs@mmmi.sdu.dk

5th Rasmus Søgaard Hansen
Department of Clinical Biochemistry
Odense University Hospital
Odense, Denmark
Rasmus.Sogaard.Hansen@rsyd.dk

6th Pernille J. Vinholt
Department of Clinical Biochemistry
Odense University Hospital
Odense, Denmark
Pernille.Vinholt@rsyd.dk

***Abstract*—Bleeding can be a life-threatening condition which occurs for 3.2% of medical patients. Information about previous bleeding and bleeding site is used to predict the risk of future bleeding and guide anticoagulant treatment. However, obtaining this information is a time-consuming task as it is contained in the free text of electronic health records.**

Previous research has mainly been focused on extracting bleeding events but does not classify the bleeding site which is important for assessing the severity of the bleeding. This study creates the first dataset for developing and evaluating machine learning models for classification of bleeding site. The dataset consists of sentences annotated by medical doctors as belonging to one of ten bleeding sites. The sentences were annotated in 149,523 electronic health record notes from 1,533 patients of Odense University Hospital, Denmark, between 2015 and 2020.

We compare different deep learning models on classifying bleeding site and find that a ~13M parameter ELECTRA model pretrained on clinical text achieves higher accuracy (0.905 ± 0.002) than a ~110M parameter general BERT model (0.884 ± 0.001) on a balanced test set of 1,500 sentences.

We furthermore test different methods for dealing with unbalanced data without finding any significant differences between methods.

***Index Terms*—Electronic health records, natural language processing, deep learning, transformer, BERT**

I. Introduction

Bleeding is a dangerous event causing increased morbidity and mortality [1], [2]. Within 14 days of admission, 3.2% of medical patients experience in-hospital bleeding, and approximately one third are considered major bleedings [3] which can be fatal.

Previous clinically relevant bleeding is a strong independent predictor for future bleeding. Bleeding history is part of the IMPROVE score [3] for determining bleeding risk in admitted medical patients and HAS-BLED [4] for addressing bleeding risk before prescription of anticoagulation therapy. The IMPROVE score defines major bleedings as those occurring within a critical organ such as intracranial, adrenal gland or spinal bleeding. Thus, information about bleeding site is an important factor for patient treatment.

Manually obtaining information about previous bleeding and its site is prohibitively time-consuming since medical doctors (MD) have to manually look through the free text of electronic health records (EHR). Automating this task would facilitate a fast way of assessing bleeding risk on an informed basis as well as decrease the time taken up by MDs manually reading through EHRs.

Several studies [5]–[7] have shown that bleeding events can be extracted from EHRs with high accuracy using deep learning techniques such as Long Short-Term Memory [8] (LSTM) networks, Convolutional Neural Networks [9] and transformer [10] variants. However, these studies do not classify the site of bleeding events. One study [11] does extract information about bleeding site using named entity recognition but does not classify the bleeding site into distinct categories.

In this paper, we create a bleeding site classification dataset using the free text of Danish EHRs. We use this dataset to train and compare an LSTM model and two transformer-based models, BERT [12] and ELECTRA [13]. We find that an in-domain ELECTRA model with ~13M parameters predicts bleeding site with an accuracy of 0.905 which is significantly better than a general domain BERT model with ~110M parameters. Moreover, we find no significant performance difference between a ~9M parameter LSTM model trained with clinical word embeddings and the BERT model. The developed models could potentially be integrated with previously developed bleeding mention detection models [5]–[7] to enrich

*Equal contribution

TABLE I
DISTRIBUTION OF THE BLEEDING SITE CLASSIFICATION DATASET.

	Train	Validation	Test	Sum
Airways	130	150	150	430
Cerebral	1,634	150	150	1,934
Ear-nose-throat	1,161	150	150	1,461
Eyes	971	150	150	1,271
Gastrointestinal	2,771	150	150	3,071
Gynecological	953	150	150	1,253
Internal bleeding	976	150	150	1,276
Skin	1,044	150	150	1,344
Urogenital	2,691	150	150	2,991
Unknown	2,003	150	150	2,303
Sum	14,344	1,500	1,500	17,344

information about bleeding risks in clinical decision support systems.

II. METHODS AND MATERIAL

A. Dataset

The dataset was created using a database of EHRs from Odense University Hospital, Denmark, between 2015 and 2020. The extracted EHRs were distributed between 13 MDs who annotated all sentences as either positive or negative for bleeding. A total of 149,523 EHR notes from 1,533 patients were annotated. This process resulted in 17,344 unique bleeding-positive sentences. Next, all the positive sentences were annotated with a consensus label from three MDs and grouped into one of ten bleeding categories: airways, cerebral, ear-nose-throat, eyes, gastrointestinal, gynecological, internal, skin, urogenital, and unknown. A preprocessing stage was done to remove superfluous spaces and special characters. After preprocessing, the sentences had an average length of 12.3 tokens.

In order to build the deep learning models, the dataset was split into train, validation, and test sets. The validation and test sets include an equal amount of sentences (n=150) from each bleeding site. The remaining were used to train the models. The dataset distribution can be seen in Table I.

B. Deep learning models

We train a LSTM model consisting of a bidirectional LSTM layer with a hidden layer size of 512. The last hidden state of the LSTM is followed by a dropout layer with probability 0.2, a dense layer of size 256, a ReLU activation function, a dropout layer of probability 0.2, and a dense classification layer. The input to the LSTM model is 300-dimensional FastText [14] word embeddings pretrained on EHRs consisting of 1.4B tokens.

Moreover, we fine-tune two transformer models, Danish BERT[1] and a locally available clinical ELECTRA model (Clin-ELECTRA). The BERT model has ~110M parameters and is pretrained on 1.6B tokens from the Danish general text domain such as CommonCrawl, Wikipedia, and Opensubtitles.

[1] Available from https://github.com/certainlyio/nordic_bert

TABLE II
SUMMARY OF DEEP LEARNING MODELS. WE = WORD EMBEDDINGS.

	Parameters	Pretrain domain	Pretrain size
LSTM	9,484,918	Clinical (WE)	1.4B tokens (WE)
BERT	110,625,034	General	1.6B tokens
Clin-ELECTRA	13,551,370	Clinical	1.4B tokens

The Clin-ELECTRA model has ~13M parameters and is pretrained on 1.4B tokens from 299,718 Danish EHRs acquired from Odense University Hospital.

We initialize the BERT and Clin-ELECTRA models from their pretrained checkpoints and follow the HuggingFace [15] implementations for text classification. Table II summarizes the deep learning models used in this study.

C. Dealing with unbalanced data

As seen in Table I, the dataset follows a natural unbalance in the amount of sentences from each bleeding site since some bleedings naturally occur more often than others. We compare three methods for training the deep learning models on the unbalanced dataset: unbalanced, upsampling, and training with a weighted loss function.

With the unbalanced method, we simply train each model with the unbalanced training distribution. With upsampling, we upsample all underrepresented classes to the size of the majority class (gastrointestinal), i.e. all categories in the training set are upsampled to include 2,771 sentences. With a weighted loss function, we assign a unique weight to each class when calculating the loss during training. The weight decreases the importance of sentences from classes with many sentences and increases the importance of underrepresented classes. Using the default in Scikit-learn [16], we calculate the weight for each class as

$$w_{c_i} = \frac{n_s}{n_c \cdot n_{s_{c_i}}} \tag{1}$$

where w_{c_i} is the weight for class i, n_s is the total number of sentences, n_c is the number of classes and $n_{s_{c_i}}$ is the number of sentences for class i. We apply w_{c_i} to the categorical cross entropy loss as

$$L = -\sum_{i=1}^{n_c} w_{c_i} \cdot y_i \cdot \log(p_i) \tag{2}$$

where L is the loss function, y_i is the true probability for class i, and p_i is the predicted probability for class i.

D. Model evaluation

We train each model multiple times with different learning rates to compare models based on confidence intervals (CI) and to report statistically significant results. Specifically, for each model we:

1) Train the model with four different learning rates and random initializations.
2) Compute the test set accuracy of the best performing model based on the loss on the validation set.
3) Repeat 1) and 2) five times.

TABLE III
ACCURACY OF THE DEEP LEARNING MODELS ON THE TEST SET USING DIFFERENT METHODS FOR TRAINING WITH UNBALANCED DATA. SE = STANDARD ERROR. CI = 95% CONFIDENCE INTERVAL. UB = UNBALANCED. US = UPSAMPLING. WL = WEIGHTED LOSS.

Model	Mean accuracy ± SE	CI
LSTM-UB	0.865 ± 0.002	0.861-0.869
LSTM-US	0.876 ± 0.002	0.871-0.881
LSTM-WL	0.881 ± 0.001	0.879-0.884
BERT-UB	0.884 ± 0.002	0.881-0.887
BERT-US	0.873 ± 0.003	0.868-0.879
BERT-WL	0.880 ± 0.001	0.878-0.882
Clin-ELECTRA-UB	0.900 ± 0.002	0.896-0.905
Clin-ELECTRA-US	0.903 ± 0.002	0.899-0.906
Clin-ELECTRA-WL	**0.905 ± 0.002**	0.901-0.910

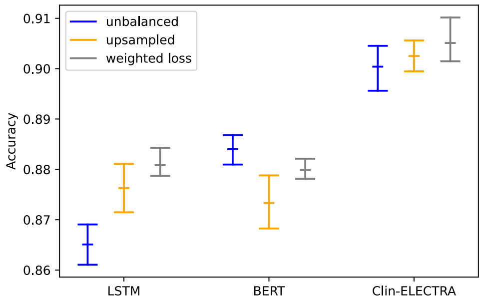

Fig. 1. Mean accuracies and 95% confidence intervals for LSTM, BERT, and Clin-ELECTRA models with the unbalanced, upsampling, and weighted loss methods.

We use the five accuracies to perform bootstrapping with 9,999 replicates to calculate mean accuracy, standard error (SE), and 95% CI for each model. Moreover, we compute the bootstrapped difference of means for each combination of models for evaluation of statistically significant differences in accuracy. For the BERT model, we search for the best model using learning rates 1e-4, 5e-5, 3e-5, and 2e-5. For Clin-ELECTRA and the LSTM model, we use learning rates 4e-4, 3e-4, 2e-4, and 1e-4. BERT and ELECTRA were trained for a maximum of 10 epochs using early stopping. The LSTM model was trained for a maximum of 50 epochs using early stopping.

III. RESULTS

Table III shows the mean accuracy, SE, and CI on the test set for all models and methods for dealing with an unbalanced dataset. Fig. 1 shows the plotted means and CIs. Table IV shows the bootstrapped CIs for difference of means for all model comparisons.

We find that Clin-ELECTRA performs significantly better than the LSTM and BERT models for both the unbalanced (0.900 ± 0.002), upsampling (0.903 ± 0.002) and weighted loss (0.905 ± 0.002) methods. We find no significant difference

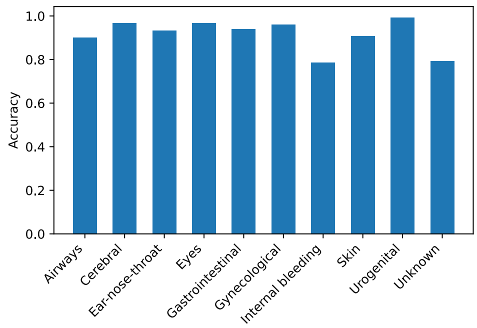

Fig. 2. Per-class accuracy for the best performing Clin-ELECTRA model (weighted loss).

between the best performing BERT model (0.884 ± 0.002) and LSTM (0.881 ± 0.001).

We do not find any of the methods for dealing with unbalanced data (unbalanced, upsampling, or weighted loss function) to perform significantly better when considering results across the three deep learning models. However, for the LSTM model, the upsampling and weighted loss methods perform significantly better than training with the unbalanced dataset. On the contrary, for BERT, the unbalanced and weighted loss methods perform significantly better than the upsampling method. For Clin-ELECTRA, we find no significant performance improvements using any of the methods.

Fig. 2 shows the per-class performance for the best performing Clin-ELECTRA model (weighted loss). It is seen that the model has the worst performance on the internal and unknown bleeding site with accuracies below 0.8. Even though only 130 airways samples are present in the training set, the model performs relatively well with an accuracy of ~90%.

IV. DISCUSSION

We found that for bleeding site classification, the ~13M parameter ELECTRA model pretrained on clinical data performs significantly better than the ~110M parameter BERT model pretrained on general data and the ~9M parameter LSTM model with clinical word embeddings. Moreover, we found no significant difference between the best BERT and LSTM model.

This result is intriguing since the computational power required for running ELECTRA and the LSTM model is much less than the BERT model. It shows that while increasing the amount of parameters has been shown to improve the performance of deep learning models [17], performance gains can also be reached by focusing on the pretraining data. Due to their pretraining domain, we hypothesize that Clin-ELECTRA and the LSTM model are able to better capture the idiosyncratic language of clinical text which differs from that of the general domain used for pretraining BERT. Clinical notes e.g. include many spelling errors, domain specific

TABLE IV

BOOTSTRAPPED 95% CONFIDENCE INTERVALS FOR DIFFERENCE OF MEANS FOR ALL MODEL COMPARISONS. MEANS ARE COMPUTED AS ROW MINUS COLUMN. * DENOTES A SIGNIFICANT DIFFERENCE AT THE 0.05 LEVEL.

	LSTM-UB	**LSTM-US**	**LSTM-WL**	**BERT-UB**	**BERT-US**	**BERT-WL**	**ELECTRA-UB**	**ELECTRA-US**	**ELECTRA-WL**
LSTM-UB	-	(-0.017, -0.005)*	(-0.021, -0.011)*	(-0.024, -0.014)*	(-0.015, -0.002)*	(-0.019, -0.010)*	(-0.041, -0.029)*	(-0.043, -0.032)*	(-0.046, -0.034)*
LSTM-US	(0.005, 0.017)*	-	(-0.010, 0.001)	(-0.013, -0.002)*	(-0.004, 0.010)	(-0.008, 0.002)	(-0.030, -0.018)*	(-0.032, -0.020)*	(-0.035, -0.022)*
LSTM-WL	(0.011, 0.021)*	(-0.001, 0.010)	-	(-0.007, 0.001)	(0.001, 0.013)*	(-0.002, 0.005)	(-0.025, -0.014)*	(-0.026, -0.017)*	(-0.030, -0.019)*
BERT-UB	(0.014, 0.024)*	(0.002, 0.013)*	(-0.001, 0.007)	-	(0.004, 0.017)*	(0.000, 0.008)	(-0.022, -0.011)*	(-0.023, -0.014)*	(-0.027, -0.016)*
BERT-US	(0.002, 0.015)*	(-0.010, 0.004)	(-0.013, -0.001)*	(-0.017, -0.004)*	-	(-0.012, -0.001)*	(-0.034, -0.020)*	(-0.035, -0.023)*	(-0.039, -0.025)*
BERT-WL	(0.010, 0.019)*	(-0.002, 0.008)	(-0.005, 0.002)	(-0.008, 0.000)	(0.001, 0.012)*	-	(-0.025, -0.016)*	(-0.026, -0.019)*	(-0.031, -0.021)*
ELECTRA-UB	(0.029, 0.041)*	(0.018, 0.030)*	(0.014, 0.025)*	(0.011, 0.022)*	(0.020, 0.034)*	(0.016, 0.025)*	-	(-0.008, 0.003)	(-0.011, 0.001)
ELECTRA-US	(0.032, 0.043)*	(0.020, 0.032)*	(0.017, 0.026)*	(0.014, 0.023)*	(0.023, 0.035)*	(0.019, 0.026)*	(-0.003, 0.008)	-	(-0.008, 0.003)
ELECTRA-WL	(0.034, 0.046)*	(0.022, 0.035)*	(0.019, 0.030)*	(0.016, 0.027)*	(0.025, 0.039)*	(0.021, 0.031)*	(-0.001, 0.011)	(-0.003, 0.008)	-

abbreviations, mix of languages (Latin, English, and Danish), and the semantic meaning of words can differ from the general domain [18].

Limitations

While this paper finds that a ~13M clinical ELECTRA model performs better than a ~110M parameter BERT model, larger general domain transformer models pretrained on more data might perform better. At the moment, no such model exists for Danish. Applying increasingly larger transformer models, however, also increases the need for expensive computational infrastructure which decreases the usability for researchers with limited access to computational power.

The clinical dataset used to pretrain the ELECTRA model could be used to further pretrain the BERT model which might improve its representations of clinical text.

The classification dataset and clinical ELECTRA model presented in this paper cannot be shared publicly due to privacy concerns but we advise interested researchers to contact us for sharing possibilities.

V. CONCLUSION

This paper created the first dataset to train machine learning models for bleeding site classification in the free text of EHRs. We find that a ~13M parameter clinical ELECTRA model can classify bleeding site with an accuracy of 0.905, surpassing both a LSTM model and a BERT model which is eight times larger in terms of parameters. Moreover, we compared different methods for dealing with unbalanced datasets. For the methods considered in this paper, we found none of them to work consistently better across all classifiers, suggesting that several approaches should be considered when working with different deep learning architectures with unbalanced data.

REFERENCES

[1] Berger JS, Bhatt DL, Steg PG, Steinhubl SR, Montalescot G, Shao M, Hacke W, Fox KA, Berger PB, Topol EJ, Lincoff AM. Bleeding, mortality, and antiplatelet therapy: results from the Clopidogrel for High Atherothrombotic Risk and Ischemic Stabilization, Management, and Avoidance (CHARISMA) trial. American heart journal. 2011 Jul 1;162(1):98-105.

[2] Cook DJ, Griffith LE, Walter SD, Guyatt GH, Meade MO, Heyland DK, Kirby A, Tryba M. The attributable mortality and length of intensive care unit stay of clinically important gastrointestinal bleeding in critically ill patients. Critical care. 2001 Dec;5(6):1-8.

[3] Decousus H, Tapson VF, Bergmann JF, Chong BH, Froehlich JB, Kakkar AK, Merli GJ, Monreal M, Nakamura M, Pavanello R, Pini M. Factors at admission associated with bleeding risk in medical patients: findings from the IMPROVE investigators. Chest. 2011 Jan 1;139(1):69-79.

[4] Pisters R, Lane DA, Nieuwlaat R, De Vos CB, Crijns HJ, Lip GY. A novel user-friendly score (HAS-BLED) to assess 1-year risk of major bleeding in patients with atrial fibrillation: the Euro Heart Survey. Chest. 2010 Nov 1;138(5):1093-100.

[5] Taggart M, Chapman WW, Steinberg BA, Ruckel S, Pregenzer-Wenzler A, Du Y, Ferraro J, Bucher BT, Lloyd-Jones DM, Rondina MT, Shah RU. Comparison of 2 natural language processing methods for identification of bleeding among critically ill patients. JAMA network open. 2018 Oct 5;1(6):e183451-.

[6] Li R, Hu B, Liu F, Liu W, Cunningham F, McManus DD, Yu H. Detection of bleeding events in electronic health record notes using convolutional neural network models enhanced with recurrent neural network autoencoders: deep learning approach. JMIR medical informatics. 2019 Feb 8;7(1):e10788.

[7] Pedersen JS, Laursen MS, Rajeeth Savarimuthu T, Hansen RS, Alnor AB, Bjerre KV, Kjær IM, Gils C, Thorsen AS, Andersen ES, Nielsen CB. Deep learning detects and visualizes bleeding events in electronic health records. Research and Practice in Thrombosis and Haemostasis. 2021 May;5(4):e12505.

[8] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997 Nov 15;9(8):1735-80.

[9] LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998;86(11);2278-2324.

[10] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Advances in neural information processing systems. 2017;30.

[11] Mitra A, Rawat BPS, McManus D, Kapoor A, Yu H. Bleeding Entity Recognition in Electronic Health Records: A Comprehensive Analysis of End-to-End Systems. AMIA ... Annual Symposium proceedings. AMIA Symposium. 2020 ;2020:860-869. PMID: 33936461; PMCID: PMC8075442.

[12] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) 2019 Jun (pp. 4171-4186).

[13] Clark K, Luong MT, Le QV, Manning CD. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. International Conference on Learning Representations 2019 Sep 25.

[14] Bojanowski P, Grave E, Joulin A, Mikolov T. Enriching word vectors with subword information. Transactions of the association for computational linguistics. 2017 Dec 1;5:135-46.

[15] Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M, Davison J. Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771. 2019 Oct 9.

[16] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J. Scikit-learn: Machine learning in Python. the Journal of machine Learning research. 2011 Nov 1;12:2825-30.

[17] Chowdhery A, Narang S, Devlin J, Bosma M, Mishra G, Roberts A, Barham P, Chung HW, Sutton C, Gehrmann S, Schuh P. Palm: Scaling language modeling with pathways. arXiv preprint arXiv:2204.02311. 2022 Apr 5.

[18] Leaman, Robert, Ritu Khare, and Zhiyong Lu. 2015. Challenges in clinical natural language processing for automated disorder normalization. Journal of Biomedical Informatics, 57:28–37.

4.4 Concluding Remarks

In this chapter, I presented the work of two papers. In the first paper, we fine-tuned the Clin-ELECTRA LLM to classify if sentences from unstructured EHR text indicate a bleeding event. The model performs with a sensitivity and specificity of 95.8% on a test set balanced on bleeding indication and location. In the second paper, we fine-tuned the Clin-ELECTRA LLM to classify the location of the bleeding from sentences indicating a bleeding event. The locations are: CNS, eye, ENT, airway, GI, internal, gynaecological, urogenital, dermatological, and unknown. The model performs with an accuracy of 90.5% on a balanced test set.

Limitations

For the bleeding event model, muscle and joint bleedings and bleedings of multiple locations could not be included in the balanced validation and test sets although the model was still trained on samples from these classes. We found relatively high sensitivity for all classes included in the test set but it was not possible to evaluate bias for these locations. This is particularly a limitation to the multiple locations class as it differs from the other classes by containing multiple descriptions of bleeding events. It is unknown how this explicitly affects the model, although the model correctly classified three out of three sentences of that class in the evaluation cohort.

It is a limitation to the bleeding location model that it does not classify muscle and joint bleedings and multiple bleedings. The limitation of not classifying multiple bleedings is mitigated by the fact that the model has the possibility to classify the location of one of the bleeding events described in the sentence. The model performs worse on internal bleedings (78.67%) and unknown bleedings (79.33%) than other locations. While samples with bleeding events of unknown location may include words such as “unknown” or “unspecified”, the lower performance for this class may be explained by samples possibly being classified based on the absence of any terms describing a location, rather than presence. The model performs worse on internal bleeding even if it has more samples (976) than three other classes—including airway bleedings with 130 samples in the training set—that all perform at or above 90% accuracy. This may be caused by internal bleeding covering more sublocations and therefore being more difficult to model than, e.g. airway bleedings.

It is noteworthy that the lack of samples of muscle and joint bleedings and sentences describing multiple bleedings are caused by a low concentration of these events in the annotation corpus—unless annotators have systematically overlooked these events. If annotators could have been provided with a corpus with a higher proportion of these bleeding locations or had time to read through a larger corpus, more samples could have been annotated. I present a method to achieve this in chapter 8. Future work includes annotating more muscle and joint bleedings and multiple bleedings.

Comparison to ICD Codes

It is not straightforward to compare the performance of the bleeding identification pipeline with the accuracy of ICD codes for bleeding. While varying results for different validations of ICD codes for bleeding were presented in section 1.1.4, the two methods of data extraction are also not directly comparable because they operate on different levels. While the ICD codes are applied on patient contact level, the pipeline is applied on sentence level, and therefore gives a more detailed output. One advantage of the pipeline is that if one sentence is predicted incorrectly, the pipeline can possibly identify the same bleeding mentioned in another sentence of the processed note. The same applies to the VTE identification pipeline presented in chapter 5.

Human in the Loop

The fact that the bleeding location model does not classify muscle and joint bleedings and mentions of multiple bleedings, and that the bleeding event model was not evaluated for bias for the same two classes, present a limitation of use of these models for studies where these specific types of bleeding need extraction. This can be managed by having a human in the loop that can validate all model predictions and correct incorrectly classified muscle and joint samples. Corrections could also include the inevitable false positive predictions caused by running a model with a specificity (and precision) below 100% on a large corpus. This is the approach used by the clinical studies that have utilised the presented bleeding identification pipeline. Although set in a different setting of reading a full EHR note, the user study of the first paper supports that MDs improve when using the model as assistance.

Application to Clinical Studies

The presented bleeding classification models have been applied to clinical research in one study and two are planned.

Hansen explored the clinical significance of bleeding in individuals with bladder cancer and used the bleeding event classification model to extract research data for determining the prevalence of bleeding before and after bladder cancer diagnosis [114]. The model identified bleeding events in a cohort of 500,000 random EHRs and positive predictions were manually validated.

In future work, the pipeline for identifying bleeding events and their location will be used for extracting research data from unstructured EHR text for a study on the bleeding incidence in admitted children. Another study will use the same pipeline to extract research data on major bleeding for a study that develops a risk score for major bleeding during admission.

CHAPTER 5

Identification of Venous Thromboembolism Events

In this chapter, I present a deep learning pipeline to automatically extract clinical research data on VTE by identifying VTE and its anatomical location in Danish unstructured EHR text. This addresses research question 2.

VTE is, like bleeding, an important parameter for clinical research, and studies would benefit from automatic extraction of research data on VTE events from unstructured EHR text. MDs in our research group defined the following relevant anatomical location categories for VTE: cerebral, eye, lung, cardiac, liver, kidney, intestines, upper extremity, and lower extremity. VTE events are included in, e.g. the IMPROVE VTE risk score [115].

The work on the models that constitute the pipeline was undertaken in the following paper:

[51] Jannik Pedersen, Martin Laursen, Pernille Vinholt, and Thiusius Rajeeth Savarimuthu. "MeDa-BERT: A medical Danish pretrained transformer model". In: *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 301–307

The models developed in this paper are the same type as those for the bleeding identification pipeline described in chapter 4. Additionally, classification of VTE events and VTE locations is not the main objective of this paper. Therefore, only the work of the paper relevant to the VTE identification pipeline is presented below. The paper is included in appendix A.

In the rest of this chapter, I first present the pre-trained LLM that was used as base for fine-tuning of the two classification models. Then, I present the VTE event classification model and VTE location classification model.

5.1 Pre-trained Large Language Model

We trained our own medical Danish BERT LLM (henceforth, MeDa-BERT) to use as base for the two models constituting the VTE identification pipeline.

The MeDa-BERT LLM was initialised from a general domain Danish BERT LLM (henceforth, Da-BERT) and further adaptively pre-trained on 134 million tokens from a Danish medical corpus that we developed. Da-BERT is a publicly available LLM which is pre-trained on 92.9 million sentences with 1.6 billion tokens from 10.7 GB Danish text from Common Crawl (9.5 GB), Danish Wikipedia (221 MB), debate forums (168 MB), and Danish OpenSubtitles (881 MB) [116].

Corpus	Type	Date retrieved	Tokens
Clinical guidelines	Guidelines	October–November 2022	80,567,576
Medicin.dk	Information portal	June 2021	28,878,335
FADL	Books	January 2022	12,531,373
Sundhed.dk	Information portal	May 2022	6,767,409
Netdoktor.dk	Information portal	October 2022	3,227,051
Wikipedia	Encyclopedia	October 2022	1,992,796
Total			**133,964,540**

TABLE 5.1: Number of tokens and date retrieved for each data source of the Danish medical corpus. This table is reproduced without any changes, other than minor formatting, from our original work [51], which is licensed under a CC BY 4.0 license. The CC BY 4.0 license can be found at `https://creativecommons.org/licenses/by/4.0/`.

The adaptive pre-training of Da-BERT, resulting in MeDa-BERT, adjusted the representations of Da-BERT to better represent the domain-specific language without pre-training from scratch [117]. The Danish medical corpus consists of data from clinical guidelines, medical information portals, books, and Wikipedia. Table 5.1 details the corpus.

5.2 Classification of Venous Thromboembolism Events and Locations

The pipeline for identification of VTE events and their location consists of two classification models. The first part of the pipeline is a VTE event classification model. It takes sentences from unstructured EHR text as input and predicts if it indicates VTE. It was trained by fine-tuning MeDa-BERT on a balanced dataset of 22,528 sentences. The second part of the pipeline is a VTE location classification model. It takes sentences positive for VTE as input and outputs the location as either lower extremity, lungs, or unknown. It was trained by fine-tuning MeDa-BERT on a dataset of 6,000 sentences. The label distributions in the datasets are shown in table 5.2.

We first presented the VTE event and VTE location datasets in earlier work [54]. Since, further annotation work was performed, and the expanded datasets were used for this paper.

Dataset	Label	Train	Validation	Test
VTE	Positive	9,064	1,100	1,100
	Negative	9,064	1,100	1,100
VTE location	Lower extremity	1,600	200	200
	Lungs	1,600	200	200
	Unknown	1,600	200	200

TABLE 5.2: Venous thromboembolism (VTE) binary event and multi-class location datasets. This table is a reproduction of a part of a table from our original work [51], which is licensed under a CC BY 4.0 license. I only include the rows describing the datasets for VTE events and VTE locations ("site" in original paper). I corrected a mistake in the original table by changing "airways" to "lower extremity". The CC BY 4.0 license can be found at `https://creativecommons.org/licenses/by/4.0/`.

The datasets were constructed from 94,520 EHR notes from Odense University Hospital and samples were labelled as the consensus of three MDs.

The paper contributes to this dissertation with a pipeline to identify VTE events and their location. The VTE event classification model has a mean balanced test set accuracy of 91.7%. The VTE location classification model has a mean balanced test set accuracy of 95.8%.

5.3 Concluding Remarks

In this chapter, I presented the work of a paper that resulted in a VTE event classification model with 91.7% accuracy and a VTE location classification model that classifies lower extremity, lungs, and unknown locations with 95.8% accuracy.

Limitations

Only the lower extremities, lungs, and unknown locations had enough samples to include them in the balanced dataset used to fine-tune MeDa-BERT for VTE location classification. This leaves cerebral, eye, cardiac, liver, kidney, intestines, and upper extremity locations unable to be classified. This is a limitation to the applicability of the VTE location model to clinical studies. More annotation is needed to gather samples for the remaining locations. I present a method for more efficient annotation in chapter 8, which can facilitate the annotation of more samples.

The paper did not explore using Clin-ELECTRA as base for the fine-tuning. Having been pre-trained on clinical EHR text as described in chapter 4, it might have represented the samples better than MeDa-BERT that was trained on medical data such as books and guidelines. There is a semantic overlap between the clinical and medical domains, but clinical data contain, e.g. spelling mistakes, abbreviations, and a different writing style than, e.g. medical books—Clin-ELECTRA should capture this better than MeDa-BERT. Clin-ELECTRA was not explored as base for fine-tuning in this work as the focus was on using LLMs that could be made publicly available. Future work includes exploring if Clin-ELECTRA can improve performance of the models.

CHAPTER 6

Extraction of Medical History

In this chapter, I present a system for automatic extraction of relevant parameters from the medical history in the form of clinical events, their attributes, and relations. This addresses research question 3.

The EHR contains important information on patients' medical history such as diagnoses, medications, treatments, and test results [2]. Information on the medical history is valuable as source of clinical research data—e.g. as support of scientific hypothesis generation. Extracted information can grow the knowledge base and increase the amount of data available for analysis. This can lead to new discoveries by inspiring research questions that can be tested in subsequent studies [118, 119]. An estimated 80% of EHR data are unstructured and infeasible to extract manually [4, 67]. Therefore, technologies to extract all information on the medical history from unstructured clinical text is needed.

Extracting and visualising the events of the medical history and their relations helps human comprehension over reading the full EHR text where the amount of information can be overwhelming. Simplified extractions, such as highlights or summaries of the unstructured text, can give new perspectives and facilitate a better overview and interpretation of potential patterns in the medical history, e.g. between two diseases or a disease and a symptom, and thereby support hypothesis generation for clinical research [67, 120]

Apart from scientific discovery, a simple overview of the medical history can aid healthcare staff in providing better treatment in clinical practice [67]. It can also aid administrative tracking of hospital data to explore trends in, e.g. diagnoses over time [119].

In the rest of the chapter, I introduce NER and RE as a means of extracting the medical history of patients from the unstructured text of the EHR, and present our work on the subject.

6.1 Named Entity Recognition and Relation Extraction

NER and RE are methods in NLP for recognition and classification of named entities and the relations between them in unstructured text. Clinical NER and RE can be used to extract important information from the medical history in the form of clinical events such as diseases, symptoms, and treatments, and the relations between them such as disease–symptom or disease–treatment relations [67].

A NER model recognises mentions of predefined semantic types, i.e. named entities, in unstructured text and classifies them into their predefined classes. NER involves two subtasks: boundary detection of the named entity, and type identification [121]. Formally, given a sequence of N tokens, $s = [w_1, \dots, w_N]$, a NER model outputs a list of tuples, $[I_{start}, I_{end}, t]$, each index in the tuples defining a named entity together. $I_{start} \in [1, N]$ and $I_{end} \in [1, N]$ are the start and end indices, i.e. the boundary, of a named entity, and t is the entity type from a predefined set of categories [121].

RE is used for classifying relations between the entities found through NER [67]. Formally, a RE model, where a relation has a direction and can consist of only two entities, takes a sequence of N entities, $s = [e_1, \dots, e_N]$, and outputs a list of tuples, $[e_{subject}, e_{object}, t]$. Each index in the tuples defines a relation between two named entities—the subject and the object. The relation type, t, is from a predefined set of categories [122].

Many different approaches have been applied to NER and RE, including deep learning methods using CNNs, RNNs, and combinations [67, 123]. Recently, transformer models have been applied to the problem with success. The Princeton University Relation Extraction system (PURE)—the state of the art on NER and RE in 2021—is an example of a transformer-based approach to NER and RE [122].

6.2 Danish Clinical Named Entity Recognition and Relation Extraction

This section presents how we adapted and trained PURE to extract clinical events, their attributes, and relations from Danish unstructured EHR text. The work is detailed in the following paper:

[48] Martin Laursen, Jannik Pedersen, Rasmus Hansen, Thiusius Rajeeth Savarimuthu, and Pernille Vinholt. "Danish Clinical Named Entity Recognition and Relation Extraction". In: *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 655–666

Below, I briefly present the contents of the paper in the context of this dissertation. The paper is included at the end of this section. It is a reprint of our original paper [48], which is licensed under a CC BY 4.0 license. The CC BY 4.0 license can be found at `https://creativecommons.org/licenses/by/4.0/`.

In this paper, we used and adapted PURE for NER and RE in Danish unstructured EHR text. PURE has separate models for NER and RE and they are described in detail in the paper. We used PURE with a Clin-ELECTRA base to extract clinical events, attributes of the events, and relations between clinical events. Clinical events were: diseases, symptoms including abnormal findings, diagnostics, treatments, anatomies including body fluids and excrements, and results. Attributes were: prior, current, future, doubt, negation, and non-patient. Relations were: "has location", "is treated with", and "has result". The dataset for training, validating, and testing the system contained 54,631 clinical events, 41,954 attributes, and 14,604 relations.

The paper contributes to this dissertation with a system for extraction of relevant parameters from the medical history. The system includes a clinical event extraction model with micro F1 63.22% (macro F1 60.05%), an attribute extraction model with micro F1 66.04% (macro F1 44.85%), and a RE model with micro F1 74.23% (macro F1 70.64%). They are all based on the PURE architecture. The clinical event extraction model has accuracies between 53% and 69% on six event types, the attribute extraction model has accuracies between 23% and 84% on six attribute types, and the RE model has accuracies between 62% and 93% on three relation types.

Danish Clinical Named Entity Recognition and Relation Extraction

Martin Sundahl Laursen[*]
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
msla@mmmi.sdu.dk

Jannik Skyttegaard Pedersen[*]
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
jasp@mmmi.sdu.dk

Rasmus Søgaard Hansen
Department of Clinical Biochemistry
Odense University Hospital

Thiusius Rajeeth Savarimuthu
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Pernille Just Vinholt
Department of Clinical Biochemistry
Odense University Hospital

Abstract

Electronic health records contain important information regarding the patients' medical history but much of this information is stored in unstructured narrative text. This paper presents the first Danish clinical named entity recognition and relation extraction dataset for extraction of six types of clinical events, six types of attributes, and three types of relations. The dataset contains 11,607 paragraphs from Danish electronic health records containing 54,631 clinical events, 41,954 attributes, and 14,604 relations. We detail the methodology of developing the annotation scheme, and train a transformer-based architecture on the developed dataset with macro F1 performance of 60.05%, 44.85%, and 70.64% for clinical events, attributes, and relations, respectively.

1 Introduction

Electronic health records (EHR) contain important information regarding the patients' medical history including diagnoses, medications, treatment plans, allergies, and test results. However, much of this information is stored in unstructured narrative text. While this information could be used to guide diagnostic decision making and treatment plans, the unstructured format makes it infeasible to fully exploit in clinical practice and research.

Natural language processing (NLP) algorithms could be used to transform the unstructured narrative text of the EHR into structured information and give medical doctors (MD) a fast overview of even a medical history spanning multiple years. NLP models' ability to process and extract information from written text keeps improving with benchmark-breaking models being published on a regular basis. For example, transformer-based models such as GPT-3 (Brown et al., 2020), BERT (Devlin et al., 2019), and ELECTRA (Clark et al., 2020) have recently shown promising results for many NLP tasks, e.g. named entity recognition and relation extraction (NER). In NER, models are trained to tag words with predefined entities and find the relations between them. In clinical NER, entities such as diseases, treatments, drugs, and tests have been extracted automatically from EHRs. However, many of the developed datasets are only in English and for specific clinical specialities or note types (Uzuner et al., 2007, 2010; Bethard et al., 2016).

This paper describes the methodology for developing the first Danish clinical NER dataset. The dataset consists of text paragraphs from Danish EHRs spanning multiple departments and note types.

First, the paper describes the clinical dataset, the strategy for choosing entities tailored to extract important information from EHRs, and the annotation scheme. Next, we train a transformer-based architecture on the developed NER dataset.

2 Related works

The annotation schemes and extracted clinical entities and relations vary. Agrawal et al. (2022) extracted medications, their status (active, discontinued, neither), and attributes. The i2b2 2009 challenge (Uzuner et al., 2010) and n2c2 2018 Track 2 (Henry et al., 2020) only extracted medications

[*]Equal contribution

and their attributes. Examples of attributes are name, dosage, mode of administration, frequency, duration, reason, strength, form, and adverse drug effects.

SemEval-2016 Task 12 (Bethard et al., 2016) extracted time entities; event entities and their contextual modality, degree, polarity, and type; and temporal relations between time and event entities (before, overlap, before—overlap, after).

SemEval-2015 Task 14 (Elhadad et al., 2015) and CLEF eHealth 2013 Task 1 (Pradhan et al., 2015) extracted disorder mentions and mapped them to their UMLS/SNOMED concept unique identifier. The former also classified attributes such as the disorder's subject, course, body location, and severity, and whether it was negated, uncertain, conditional, or generic.

The i2b2 2010 challenge (Uzuner et al., 2011) extracted entities (medical problems, treatments, tests), assertions (present, absent, possible, conditional, hypothetical future, and associated with someone other than the patient), and relations between medical problem entities and each of medical problem, treatment, and test entities.

The i2b2 2012 challenge (Sun et al., 2013b) extracted clinically relevant events. Their type was classified as concept (problem, test, treatment), clinical department, evidentials indicating source of information, or occurrences (events that happen to the patient). Polarity was classified as positive or negated, and modality as happens, proposed, conditional, or possible. Temporal expressions were extracted with their type (date, time, duration, frequency), value, and modifier indicating whether the temporal expression was exact or not. Temporal relations indicating the type of connections between events and temporal expressions were also extracted.

3 Methods

This section describes the data, annotation scheme, and model used for Danish clinical NER.

3.1 Data

We extracted 11,607 paragraphs with a length between 11 and 75 words from EHRs from Odense University Hospital in Denmark. Paragraphs were sampled randomly from different EHR note types across every department of the hospital to ensure the data distribution would resemble that of EHRs: 46% were from clinical contacts, 13% primary journals, 10% care data, 3% epicrises, 3% ambulatory care contacts, 2% surgical notes, 2% emergency room journals, and 20% were from 55 different minor EHR note types. Paragraphs were lowercased and anonymised by two of the authors.

Clinical event	Description
Disease	A disorder of structure or function, especially one that has a known cause and a distinctive group of symptoms, signs, or anatomical changes. Examples include cancer, influenza, and narcolepsy.
Symptom	A symptom is a physical or mental feature which is regarded as indicating a condition of disease, particularly such a feature that is apparent to the patient. We include abnormal findings, which the MD makes when examining the patient objectively, as these are sometimes coinciding with symptoms—e.g. bruises. Examples include headache, stomach ache, and pain.
Diagnostic	Any tool or method concerned with the diagnosis of illnesses or other problems. Includes measurements and tests. Examples include CT scans, blood samples, and temperatures.
Treatment	A treatment is any medical care given to a patient for an illness or injury. Examples include medication, plaster, and rehabilitation.
Anatomy	Any part of human anatomy. Includes body fluids and excrements. Examples include arms, organs, and blood.
Result	All results of diagnostics that do not carry any meaning without being coupled to the diagnostic. Examples include numbers that indicate length, temperature, or volumes. Diseases or symptoms found by diagnostics are annotated as such, e.g. a tumour found by a CT scan.

Table 1: Description of clinical events. Descriptions were inspired by the Oxford English Dictionary.

3.2 Annotation

3.2.1 Annotation scheme

Two MDs with expert clinical domain knowledge developed the annotation scheme through an iterative process of making annotation rules and testing them.

Annotation rules were made to extract clinically relevant information from the medical history. Focus was for the rules to be as complete as possible to capture all important information about the medical history while still being simple to use for the annotators.

We extracted three types of information: clinical events, the attributes of the clinical events, and relations between the clinical events.

Clinical events were: diseases; symptoms, including abnormal findings; diagnostics; treatments; anatomies including body fluids and excrements; and results. Symptoms and abnormal findings were joined in one as they sometimes coincided. Normal findings were not included as there were so many that they would cloud the visualisation of the history. Table 1 shows all clinical events and their descriptions as defined by the medical experts.

Clinical events were further described by their attributes. Attributes were: prior; current; fu-

Attributes	Description
Prior	Entities that occurred in prior admissions or in the distant past. Includes treatments that are being stopped at that point in time.
Current	Entities that occur in the present. Includes prescribed medicine.
Future	Entities that occur or might occur in the future—e.g. the risk of skin cancer, or ordering diagnostics for a later day.
Doubt	Any entity that is not confirmed. Includes any treatments that might need to be started in the future.
Negation	Entities such as diseases or symptoms that are mentioned as not being present.
Non-patient	Entities that are not related to the patient in question. One example is the disease history of the patient's relatives.

Table 2: Description of attributes.

ture; doubt; negation; and non-patient. All clinical events could take one of the six attributes except anatomies and results. Anatomies did not take any attributes while results could only take a prior or current attribute. Table 2 shows all attributes and their descriptions.

Clinical events could connect to each other in limited ways through one-way relations. Diseases, diagnostics, and symptoms could connect to anatomies through a "has location" relation. Diseases, symptoms, and anatomies could connect to treatments through a "is treated with" relation. Diagnostics could connect to results through a "has result" relation.

Figure 1 shows an overview of the clinical events, attributes, and relations. Appendix A shows the full annotation guidelines with further details and explanations to the annotators.

3.2.2 Annotation process

Six annotators were recruited for the task. Five were Master of Science in Medicine students and one was a MD.

Figure 2 shows the process of annotator training. It included reading the annotation guide and an iterative process of annotating a learning set of 55 paragraphs (not included in dataset) followed by error analysis until a final test was made on a set of 98 gold paragraphs annotated by an expert MD. Paragraphs were annotated using the CLAMP software (Soysal et al., 2017). We report the micro F1 of each annotator on the gold set.

Figure 3 shows an example of an annotated paragraph.

3.3 Entity and relation extraction model

This section describes the architecture of the Princeton University Relation Extraction system (PURE) (Zhong and Chen, 2021) which we used and adapted for Danish clinical NER. It further describes the dataset used and the training of the models.

3.3.1 Model architecture

PURE—the 2021 state-of-the-art on entity and relation extraction—is a NER deep learning model based on a transformer structure. The model has a separate entity and relation extraction part.

For entity extraction, the model takes as input all possible text spans up to a maximum length. A transformer extracts contextual word embeddings for the start and end token of each span. They are concatenated with a learned span width embedding and classified by a feedforward network.

When extracting relations, for each candidate pair of entities, the text is passed through a transformer with inserted entity start and end marker tokens for the subject and object entity, also indicating the type. The concatenation of the start marker token for the candidate subject and object entity is classified by a feedforward neural network.

We used PURE's entity extraction approach for clinical events and the relation extraction approach for relations between clinical events.

We used our own approach adapted from the PURE relation extraction approach for attributes. We inserted clinical event start and end marker tokens, passed all tokens through a transformer, concatenated the start and end marker tokens, and classified the attribute using a feedforward network. The marker tokens were used for classification instead of the word(s) forming the clinical event to guide the model to look more at the context rather than the specific word—the context being the important factor in attribute classification. Additionally, enriching the input with the type of the clinical event could guide the model if attributes were described differently for different clinical events.

Figure 4 shows the three types of extraction tasks.

3.3.2 Datasets

Table 3 shows the number of clinical events, attributes, and relations by type in the train, validation, and test set. The dataset had a total of 11,607 paragraphs, each containing a varying number of clinical events, attributes, and relations. On average, each paragraph contained 4.7 clinical events, 3.6 attributes, and 1.3 relations. We split the paragraphs in train, validation, and test sets for an approximate 80%–10%–10% ratio between each type of clinical event, attribute, and relation. The sets were unbalanced on type of entity or relation—e.g. for the attributes training

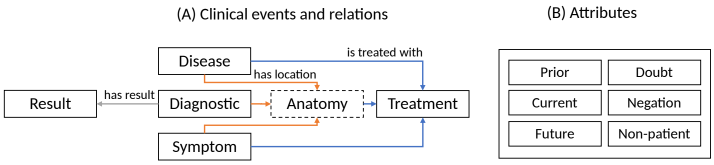

Figure 1: (A) Clinical events and relations between them. Symptoms include abnormal findings. Anatomies include body fluids and excrements. Diagnostics include measurements and tests. Blue: "is treated with". Orange: "has location". Grey: "has result". (B) Attributes. Anatomy (dashed lines) takes no attributes. Other clinical events must take one attribute. Results only take prior or current attributes.

	Train (% of row total)	**Validation (% of row total)**	**Test (%of row total)**	**Total (% of column total)**
Paragraphs	9,687 (83%)	960 (8%)	960 (8%)	**11,607 (100%)**
		Clinical events		
Diseases	2,033 (78%)	295 (11%)	272 (10%)	**2,600 (5%)**
Symptoms	11,937 (80%)	1,455 (10%)	1,571 (10%)	**14,963 (27%)**
Diagnostics	8,921 (80%)	1,095 (10%)	1,194 (11%)	**11,210 (21%)**
Treatments	6,918 (79%)	911 (10%)	882 (10%)	**8,711 (16%)**
Anatomies	10,172 (80%)	1,227 (10%)	1,278 (10%)	**12,677 (23%)**
Results	3,522 (79%)	473 (11%)	475 (11%)	**4,470 (8%)**
TOTAL	**43,503 (80%)**	**5,456 (10%)**	**5,672 (10%)**	**54,631 (100%)**
		Attributes		
Prior	2,028 (80%)	237 (9%)	283 (11%)	**2,548 (6%)**
Current	23,217 (79%)	3,021 (10%)	3,109 (11%)	**29,347 (70%)**
Future	1,237 (79%)	161 (10%)	160 (10%)	**1,558 (4%)**
Doubt	2,479 (82%)	263 (9%)	289 (10%)	**3,031 (7%)**
Negation	3,890 (80%)	496 (10%)	500 (10%)	**4,886 (12%)**
Non-patient	480 (82%)	51 (9%)	53 (9%)	**584 (1%)**
TOTAL	**33,331 (79%)**	**4,229 (10%)**	**4,394 (10%)**	**41,954 (100%)**
		Relations		
is treated with	1,485 (80%)	175 (9%)	197 (11%)	**1,857 (13%)**
has location	6,501 (80%)	779 (10%)	823 (10%)	**8,103 (55%)**
has result	3,652 (79%)	499 (11%)	493 (11%)	**4,644 (32%)**
TOTAL	**11,638 (80%)**	**1,453 (10%)**	**1,513 (10%)**	**14,604 (100%)**

Table 3: Composition of the train, validation and test sets by type of clinical event, attribute, and relation.

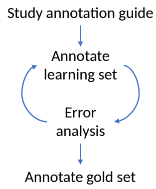

Figure 2: Annotator training process. Figure inspired by Sun et al. (2013a).

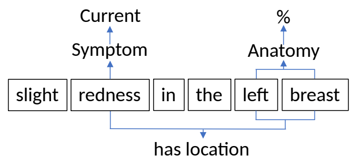

Figure 3: Example of annotated paragraph. % signifies that no attribute could be assigned to the clinical event per the annotation scheme.

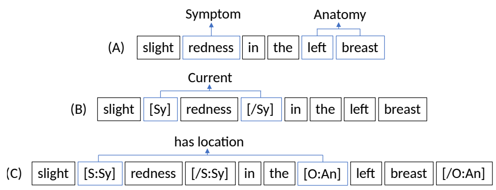

Figure 4: (A) Classification of clinical events from start and end tokens of span. Span width embedding not depicted. (B) Classification of attribute using clinical event marker tokens. (C) Classification of relation using subject/object and clinical event marker tokens. Figure inspired by Zhong and Chen (2021).

Evaluation metric	Loss	Micro R%	Micro P%	Micro F1%	Macro R%	Macro P%	Macro F1%
Micro F1	Unweighted	79.14	79.14	79.14	38.34	40.51	38.56
	Weighted	61.81	61.81	61.81	45.35	33.20	34.23
Macro F1	Unweighted	77.30	77.30	77.30	41.88	41.90	41.48
	Weighted	60.13	60.13	60.13	51.37	41.87	43.85

Table 4: Validation set micro and macro recall, precision, and F1 score on the attribute extraction task when selecting the best iteration of the model based on micro and macro F1 score with unweighted and weighted loss. 2 hidden layers of size 75 was used for the test. R: Recall. P: Precision.

set, there were 23,217 current and only 480 non-patient attributes. All datasets were in the json format used by PURE (see Zhong and Chen (2021)).

3.3.3 Training

When training the clinical event extraction model, we used a Danish Clinical ELECTRA pretrained on the narrative text from 299,718 EHRs from Odense University Hospital as the transformer base (Pedersen et al., 2022). The model had $\sim$13M parameters and consisted of 12 transformer layers with 4 attention heads. We used a dropout of 0.1 after the last ELECTRA hidden layer output. We tested classification heads with two hidden layers of varying size, each followed by a dropout of 0.2 and a ReLU activation function. We used a maximum span of 8 and a train batch size of 32. We trained for 100 epochs using the AdamW optimizer with learning rate 1e-5 for the transformer layers and 1e-4 for the classification head, and a warm-up proportion of 0.1.

When training each of the models for extracting attributes and relations, we used the same transformer base with a normalisation layer and a dropout of 0.1 after the concatenation of tokens. We tested classification heads with two hidden layers of varying size, each followed by a dropout of 0.2 and a ReLU activation function. We further tested a classification head only consisting of a single classification layer. We used a train batch size of 32 and a maximum sequence length of 128. We trained for 20 epochs using the AdamW optimizer with learning rate 2e-5 and a warm-up proportion of 0.1.

We modified the training method of PURE to guide the models towards equal performance on all classes by using a weighted loss function to counteract the unbalanced dataset and chosing the best model for each of the clinical event, attribute, and relation extraction tasks as the model iteration with the best macro F1 on the validation set, rather than the micro F1 standard of PURE. Table 4 shows a test of the performance on the attribute extraction task when selecting the best iteration of the model based on micro and macro F1 score with unweighted and weighted loss. Using the macro F1 score with weighted loss gave the best performance across all classes. Appendix B shows the confusion matrices for each combination.

Class weights were calculated for the training of each model using the default formula in Scikit-learn (Pedregosa et al., 2011):

$$w_x = \frac{n_{samples}}{n_{classes} \cdot n_x} \quad (1)$$

where x is the class, $n_{samples}$ is the number of total samples, and $n_{classes}$ is the number of classes. The negative class, i.e. samples not to be given any label by the model, was given a weight of 1.

The negative class was excluded when calculating the F1. We only trained the attribute and relation models to make classifications that were allowed for the connected clinical events according to the annotation scheme. Appendix C shows the results of the hyperparameter search. We report the micro and macro recall, precision, and F1 for the best models on the test set.

4 Results

This section presents the agreement of the annotators on the gold set and the results of the Danish clinical NER models.

4.1 Annotation

Table 5 shows the annotators' micro F1 performance on the gold set. For clinical events, it ranged 83.71%–91.24% (average 85.62%) for overlapping matches, and 74.12%–85.15% (average 77.67%) for exact matches. For attributes, it ranged 79.21%–86.19% (average 81.71%) and for relations 71.28%–90.06% (average 77.79%).

4.2 Entity and relation extraction model

The models that had the best validation performance in the hyperparameter search were:

- A clinical event extraction model with two hidden layers of size 450 in the classification head.
- An attribute extraction model with a single classification layer.
- A relation extraction model with two hidden layers of size 150 in the classification head.

Annotator	**A**	**B**	**C**	**D**	**E**	**F**
	Overlap match, micro F1%					
Clinical event	91.24	84.22	84.41	85.71	84.43	83.71
Attribute	86.19	83.06	79.21	81.29	79.75	80.75
Relation	90.06	76.97	75.60	77.01	71.28	75.84
	Exact match, micro F1%					
Clinical event	85.15	76.08	76.29	78.69	74.12	75.71

Table 5: The anonymised annotators' performance on the gold set. Exact match: a match is defined as the exact tokens annotated in the gold set with the same label. Overlap match: a match is defined as minimum one token overlapping with the gold set annotation of the same label. Only an overlap match F1 is calculated for attributes and relations as evaluating an exact match would propagate the potential error in the span of the clinical event to which the attribute or relation is connected.

	Micro			**Macro**		
	R%	**P%**	**F1%**	**R%**	**P%**	**F1%**
	Overlap match					
Clinical events	66.29	77.31	71.38	64.88	72.60	68.20
	Exact match					
Clinical events	60.97	65.64	63.22	59.84	61.30	60.05
Attributes	66.04	66.04	66.04	51.60	42.64	44.85
Relations	75.88	72.66	74.23	74.74	67.85	70.64

Table 6: Performance of the best clinical event, attribute, and relation extraction models on the test set. Attributes and relations are only reported with an exact match as the models do not consider the span of the clinical event from which the attribute or relation is classified. R: Recall. P: Precision.

Table 6 shows the performance of the best models on the test set. Clinical events were extracted with exact micro F1 63.22% and macro F1 60.05%, attributes with micro F1 66.04% and macro F1 44.85%, and relations with micro F1 74.23% and macro F1 70.64%. The negative class was excluded when calculating the recall, precision, and F1 scores.

Figure 5 shows the confusion matrices of performance on clinical events, attributes, and relations. The confusion matrices include the clinical events and relations that were not extracted and falsely extracted by the model ('O').

The model for clinical event extraction performed best on anatomies (69%) and worst on results (53%). 1,568 spans were falsely extracted

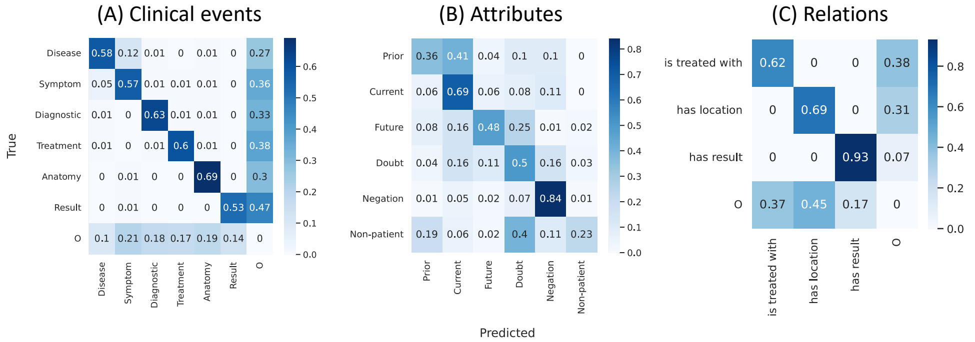

Figure 5: Confusion matrices of performance on (A) clinical events, (B) attributes, and (C) relations. 'O' counts the clinical events and relations that were not extracted and falsely extracted by the model.

as a clinical event with symptoms being the most frequent (21%). The model for attribute extraction performed best on negations (84%) and worst on non-patient (23%). The model for relation extraction performed best on "has result" (93%) and worst on "is treated with" (62%). 432 false relations were extracted of which "has location" was the most frequent misclassification (45%).

5 Discussion and limitations

This paper presented a methodology for developing a dataset for Danish clinical NER. It presented an annotation scheme for annotation of all clinical events, their attributes, and relations that are relevant for the medical history. The dataset included text paragraphs from Danish EHRs spanning multiple departments and note types.

We trained and adapted PURE NER deep learning models to extract clinical events (overlap match macro F1 68.20%; exact match macro F1 60.05%), attributes of clinical events (macro F1 44.85%), and relations between clinical events (macro F1 70.64%). The results are promising for Danish clinical NER but need improvement. A discussion of possible improvements to the methodology, limitations, and future work is provided below.

The clinical event extraction model had similar performance on all classes with accuracies between 53% (results) and 69% (anatomies). There was little contamination between classes as most errors were caused by failure to extract or false extraction of a clinical event. There was some contamination between symptoms and diseases with 12% of diseases being classified as symptoms and 5% of symptoms being classified as diseases. This supports claims by annotators that diseases and symptoms in some cases are difficult to differentiate and that extra attention must be given to differentiate these in the annotation guidelines.

The attribute extraction model had large differences in performance with accuracies between 23% (non-patient) and 84% (negation). There were more misclassifications of the non-patient attribute as doubt (40%) than correct classifications. The future and doubt attributes had significant contamination between them with 25% and 11% misclassifications as the other class, respectively. The many misclassifications between non-patient and doubt attributes, and especially future and doubt attributes, could indicate that the model would improve if the non-patient, doubt, and future attributes were merged to a single class of uncertain attributes. This would most likely not harm the usefulness of the model to MDs significantly.

The fact that more prior attributes were misclassified as current (41%) than correct classifications (36%) likewise indicates that these two attributes could be merged into a single class of clinical events that occurred. This would, however, decrease the usefulness of the model as it is important for MDs reviewing the medical history to know if a clinical event is prior or current.

The relation model extracted 93% of the "has result" relations, and 62% and 69% of the "is treated with" and "has location" relations, respectively. The differences are likely caused by the fact that the "has result" relation only connects diagnostics to results while the two other relations have three different one-way relationships.

In this paper, we only explored one type of NER model and tested a limited set of architectures and hyperparameters. Future work could include testing other architectures and enriching the model input with more information, e.g. the output of a text parser, which could help differentiate attributes dealing with the time-aspect. The six annotators had an average micro F1 (overlap match) of 85.62%, 81.71%, and 77.79% for clinical events, attributes, and relations, respectively. Merging certain attributes and more emphasis on differences between symptoms and diseases could increase these scores.

The Danish clinical NER dataset is not made publicly available due to it containing sensitive information. We advise interested researchers to contact us for sharing possibilities.

6 Conclusions

This paper presented methodology and annotation scheme for developing the first Danish clinical NER dataset. The corpus consists of 11,607 paragraphs annotated for six entity types, six attributes, and three relations. The corpus was used to fine-tune language models which showed promising results for classifying the entities, attributes, and relations of the dataset.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 1052–1062.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. https://openreview.net/forum?id=r1xMH1BtvB Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. https://doi.org/10.18653/v1/N19-1423 BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Noémie Elhadad, Sameer Pradhan, Sharon Gorman, Suresh Manandhar, Wendy Chapman, and Guergana Savova. 2015. Semeval-2015 task 14: Analysis of clinical text. In *proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 303–310.

Sam Henry, Kevin Buchan, Michele Filannino, Amber Stubbs, and Ozlem Uzuner. 2020. 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. *Journal of the American Medical Informatics Association*, 27(1):3–12.

Jannik S Pedersen, Martin S Laursen, Cristina Soguero-Ruiz, Thiusius R Savarimuthu, Rasmus Søgaard Hansen, and Pernille J Vinholt. 2022. Domain over size: Clinical electra surpasses general bert for bleeding site classification in the free text of electronic health records. In *2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 1–4. IEEE.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Sameer Pradhan, Noémie Elhadad, Brett R South, David Martinez, Lee Christensen, Amy Vogel, Hanna Suominen, Wendy W Chapman, and Guergana Savova. 2015. Evaluating the state of the art in disorder recognition and normalization of the clinical narrative. *Journal of the American Medical Informatics Association*, 22(1):143–154.

Ergin Soysal, Jingqi Wang, Min Jiang, Yonghui Wu, Serguei Pakhomov, Hongfang Liu, and Hua Xu. 2017. https://doi.org/10.1093/jamia/ocx132 CLAMP – a toolkit for efficiently building customized clinical natural language processing pipelines. *Journal of the American Medical Informatics Association*, 25(3):331–336.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013a. https://doi.org/https://doi.org/10.1016/j.jbi.2013.07.004 Annotating temporal information in clinical narratives. *Journal of Biomedical Informatics*, 46:S5–S12. Supplement: 2012 i2b2 NLP Challenge on Temporal Relations in Clinical Data.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013b. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Özlem Uzuner, Yuan Luo, and Peter Szolovits. 2007. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

Özlem Uzuner, Imre Solti, and Eithon Cadag. 2010. Extracting medication information from clinical text. *Journal of the American Medical Informatics Association*, 17(5):514–518.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61.

Appendices

A Annotation guidelines

A.1 Clinical events

A.1.1 Disease

Contains all diseases including diseases that could be considered a result of a Diagnostic.

A.1.2 Symptom

Includes all symptoms and abnormal findings. Findings that are not abnormal should not be annotated. However, a negation of an abnormal finding should be annotated because the abnormal finding is mentioned even though it is not present. For example, “fracture” should be annotated in the sentence “there is no sign of fracture.”

If there is a negation of a non-abnormal finding, it should be annotated in the entity. For example, “cannot hear” is annotated in the sentence “patient cannot hear anything.”

In the sentence “no symptoms,” the word “symptoms” should not be annotated as a symptom, as it does not contain any information.

In case a symptom or abnormal finding is found by a Diagnostic, there may be a coincidence with the Result entity. Here, it is annotated as Symptom if the entity can provide sufficient meaning alone. For example, “cyst” or “tumour.”

If the Symptom cannot stand alone and one needs to know which Diagnostic was carried out in order to understand the result, the entity should instead be annotated as Result and have a “has result” relationship from the Diagnostic entity. For example, this applies to “Temp: 24 C” and “Stix: 3+”. “Temp” and “Stix” are annotated as Diagnostic with “is treated with” relationship to Result “24 C” and “3+.”

A.1.3 Result

Includes all results of Diagnostic, e.g. values and blood test results.

A Result cannot stand on its own. A relation from the Diagnostic is needed for it to make sense. These can be entities like “stable”, “positive”, “negative”, “24 C” or “3+”.

Typically, this entity will appear in sentence structures with a colon: “Diagnostic: Result”. Note that the two entities are mentioned very close to each other in the text—in this case only with a colon in between. An example could be “Temp: 24 C” or “Stix: 3+”. “Temp” and “Stix” are annotated as Diagnostics with a “has result” relation to Result “24 C” and “3+”.

Entities that can instead be annotated as Symptom will typically be mentioned further away or completely lack a Diagnostic as a Symptom can stand alone and make sense.

See also the description for Symptom.

A.1.4 Diagnostic

Includes all diagnostics, measurements, and tests. This can include CT scans, blood tests, MR scans, and recordings of a newborn’s length, temperature, etc.

Note that “blood sample results” and “radiology description” are not a Diagnostic and should not be annotated.

If KAD is mentioned along with a volume, e.g. “KAD emptied of 200 mL,” it is marked as Diagnostic - Result. If there is no volume specified, KAD is annotated as Treatment.

A.1.5 Treatment

Includes all forms of treatment including medication.

To annotate entities as concisely as possible, for example in the sentence “good effect of 2.5 mg morphine IV,” only “morphine” should be annotated as Treatment.

In the sentence "treated for xxx," the word "treatment" should not be annotated as Treatment as it does not contain any information.

If KAD is mentioned without a volume indication, it should be annotated as Treatment. If KAD is mentioned with a volume, for example "KAD emptied for 200 mL," it should be annotated as "Diagnostic - Result."

A.1.6 Anatomy

Includes all mentions of anatomies and things from the body (blood, feces, urine, sweat, etc.).

Typically used to indicate the location of a Disease or Symptom, a Diagnostic, or a Treatment. Examples: "brain", "left foot" or "duodenum".

When Anatomy is described by an adjacent word, for example "left", this should be included in the entity.

Remember to annotate the Anatomy entities that should not be linked to other entities.

A.2 Attributes

A.2.1 Current

The entity is either present, carried out, or current. If medication is prescribed to the patient, this should also be marked as "Treatment - Current", as it can be assumed that the treatment will start and it may be the last time it is mentioned in the journal. On the other hand, "Scheduling a CT for Tuesday." should be marked as "Future" as it will be described in a future medical note, for example with the result.

A.2.2 Negation

The entity is not present. For example, if it is mentioned that the patient does not have a fracture, the fracture should be marked as Symptom - Negation. Note that the word "not" should not be part of the marked entity. However, if there is a negation of a normal finding, it should be annotated as such. For example, "cannot hear" in the sentence "patient cannot hear anything" is annotated as Symptom - Present.

A.2.3 Prior

If the entity refers to a previous case, i.e., a previous hospitalisation or if it happened a long time ago. For example, it should be annotated as a prior Treatment when a cast or drain is removed, as the treatment is finished. However, if a CT scan from the previous day is mentioned, it should be annotated as Current.

A.2.4 Future

Everything that takes place in the future. For example, cancer is annotated as Disease - Future if it is mentioned that "there is a risk of cancer if you use tanning beds too often."

It is marked as Diagnostic - Future if an MRI scan is planned for the next day. However, if it is written "the treatment with xxx starts" or "rp. xxx" it should be marked as Treatment - Current as it is assumed that the treatment will certainly happen.

Also includes references to possible future treatments.

A.2.5 Doubt

If the patient might have a disease that has not yet been confirmed.

If a Treatment should be given provided that certain things change.

The difference between Doubt and Future is that Future is more certain - it is going to happen - while Doubt is more uncertain or conditional.

A.2.6 Non-patient

If an entity does not have a direct connection to the patient. This can occur when a general letter is sent out regarding cancer screening. Cancer should then be annotated as Disease - Non-patient. If it is mentioned that the patient's mother had a certain disease, it should also be annotated in this way.

A.3 Relations

When entities are annotated, the relationships between entities can be annotated. This is done by pulling the "From entity" over to the "To entity". The direction of the relationship is important. Therefore, pay attention to the name of the relationship and read it out loud if necessary, "Entity - Relation - Entity" and listen to see if it makes sense or if the arrow needs to be reversed. CLAMP will show which relationships can be annotated for the pair being drawn between.

has location

From entities: Disease, Symptom, Diagnostic.
To entities: Anatomy.

has result

From entities: Diagnostic.
To entities: Result.

is treated with

From entities: Disease, Symptom, Anatomy.
To entities: Treatment.

The "is treated with" relation links the entities Disease, Symptom, and Anatomy to a Treatment. In some cases, sentences describing a required treatment could be linked to both an Anatomy and Treatment entity. In this case, the Treatment should be linked to the Symptom instead of the Anatomy. You should only link the Anatomy to the Treatment using the "is treated with" relation if the Treatment cannot be linked to anything else. Example: "Left knee skin scraping is treated with plaster." Annotation: skin scraping - "Treated with" - plaster.

A.4 General notes

It is important not to annotate periods, commas, etc. unless they are part of an abbreviation. For example, in "Patient has cancer," only "cancer" and not "cancer." should be marked. If you double-click a word, CLAMP will only mark the word and not any punctuation next to the word. This can make it a bit troublesome to include periods in abbreviations.

Entities should be annotated as concisely as possible without losing meaning. This means that in the sentence "there are signs of cancer," only "cancer" and not "signs of cancer" should be marked as an entity. If an entity has some describing words next to it, the following rule can be used to decide how much should be annotated. In the sentence "pain in the front of the arm," only "arm" is marked as Anatomy since "front" and "arm" are connected through the word "of." In the sentence "pain in the left arm," "left arm" is marked as Anatomy since there are no words between "left" and "arm". In sentences describing a prescription of medication, only the name is marked as Treatment, and not, for example, the quantity indication or the number of days.

Entities may not overlap with each other.

B Selection of loss and evaluation metric

Figure 6 shows the confusion matrices for the attribute extraction task when selecting the best iteration of the model based on micro and macro F1 score with unweighted and weighted loss.

Using the micro F1 to select the best iteration of the model resulted in some classes being practically excluded during classification. Using the macro F1 to select the best model iteration and training with a weighted loss gave the most equal performance on all classes.

	Classification head hidden layers	**Validation Exact F1 %**
Clinical event	2x 75	58.49
	2x 150	59.82
	2x 300	60.68
	2x 450	61.34
	2x 600	60.91
Attribute	None	48.01
	2x 50	43.20
	2x 75	43.85
	2x 150	44.10
	2x 300	44.32
Relation	None	66.15
	2x 75	68.39
	2x 150	68.85
	2x 300	67.39

Table 7: Results of the hyperparameter search.

C Hyperparameter search

Table 7 shows the results of the hyperparameter search.

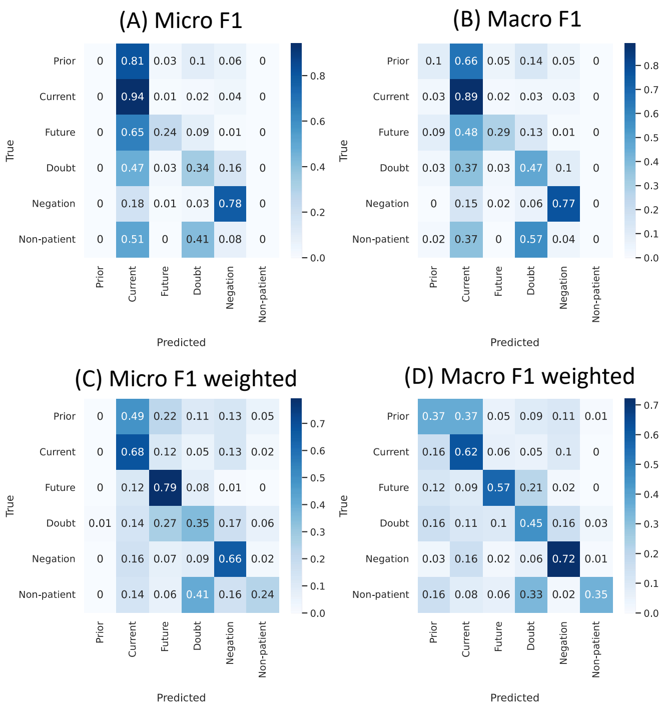

Figure 6: Confusion matrices showing the attribute extraction validation performance of the models chosen based on (A) micro F1, (B) macro F1, (C) micro F1 trained with weighted loss, and (D) macro F1 trained with weighted loss.

6.3 Concluding Remarks

In this chapter, I presented the work of a paper that trained a clinical event extraction model with micro F1 63.22%, an attribute extraction model with micro F1 66.04%, and a RE model with micro F1 74.23%. Clinical events were: diseases, symptoms including abnormal findings, diagnostics, treatments, anatomies including body fluids and excrements, and results. Attributes were: prior, current, future, doubt, negation, and non-patient. Relations were: "has location", "is treated with", and "has result".

Considerations about Annotation Scheme

The annotation scheme was decided by two MDs in an iterative process of making and testing the rules. Emphasis was on complete rules to extract relevant clinical information, but not impossible to annotate. Considerations during the iterative process, not disclosed in the paper, include splitting symptoms and abnormal findings, but it was found too difficult to differentiate them during annotation. Another consideration was to have one entity type for all findings—normal and abnormal—but we found that there were too many normal findings which made the annotation infeasibly slow.

Extraction versus Classification

The NER and RE models developed in this chapter can extract many typea of information from the medical history, but the structured output is different to that of the classification models of chapters 4 and 5. The classification models are typically used for more specific classes, such as bleeding or VTE, while NER models extract text spans and typically classify them into more broad classes, such as disease or treatment. If a NER model extracts the text span "influenza" with a "disease" type, another classification must take place to interpret the information contained in the text span—that the disease is influenza. NER is suited to capturing heterogenous information, such as the medical history where multiple classification models would have been needed to be trained.

Limitations and Future Work

All three models presented in this chapter need improved performance for clinical research to be able to take advantage of the full capabilities of the system. Training the models on more samples will most likely improve performance, but annotation for clinical NER and RE on the medical history is a particularly time-consuming and difficult process as every sample can contain many different clinical events, attributes, and relations. Multiple annotators must typically be involved, which demands very careful coordination of annotator guidelines.

Prompt-based methods like that of Liu et al. [111] presented in section 3.4 could potentially completely remove the need of an annotation process. This will be explored in future work.

CHAPTER 7

De-identification of Unstructured Electronic Health Record Text

In this chapter, I present a deep learning model trained with weak supervision for automatic de-identification of names, streets, and locations in Danish unstructured EHR text. This addresses research question 4.

As described in the introduction, EHR text must be de-identified before being used for research unless explicit consent is acquired [40]. De-identification involves the removal of all PII. While it is a relatively simple task to exclude structured data fields of the EHR that contain, e.g. name, birthday, and social security number when extracting EHR data, it is more complicated to de-identify the unstructured EHR text where identifiers may be disclosed in narrative text.

The GDPR defines personal data as, in simple terms, any information related to a person who can be identified directly or indirectly by identifiers or a combination of identifiers, such as names; identification numbers; location data; online identifiers; and factors specific to the identity of the person such as physical, mental, or social. PII can also be indirect information that can be used with additional information to identify the person [40].

The United States HIPAA privacy rule offers a more tangible guide with the Safe Harbor de-identification method, defining data as de-identified when 18 types of identifiers of the individual and its relatives, employers, and household members are removed. The identifiers include names, geographic subdivisions smaller than a state, dates, phone numbers, e-mail addresses, URLs, social security numbers, medical record numbers, and ten other identifiers [107].

While some identifiers such as dates, phone numbers, e-mail addresses, URLs, social security numbers, and medical record numbers may be removed using search patterns for numeric combinations, other identifiers like names, cities, and streets are not straightforward to remove. Even with a list of all existing names, cities, and streets, many are homographs, i.e. ambiguous because they share spelling with other words. In a clinical context, homographs are particularly challenging as some treatments and diseases are eponymous, having been given a name from the person who discovered it. Removing a treatment in the unstructured text, mistakenly believing it to be PII, is decremental to the quality of the data and therefore, clinical research. Vice versa, failing to de-identify PII does not maintain data privacy.

Machine learning methods such as transformer-based deep learning architectures that can distinguish homographs based on their context are ideal for automatic de-identification. Typically, the clinical domain requires specialised models to capture the domain-specific terms and sentence structures to achieve the best performance [42].

As Danish is a low-resource language [43], labelled datasets and trained models for de-identification of specific identifiers are lacking, and manually labelling a dataset for conventional training of models from scratch can be as time-consuming and expensive as manual de-identification. Methods that rely on transfer learning, like LLMs, and less labelled data can overcome these challenges.

Weak supervision is a semi-supervised learning method, which requires less expert-labelled data than conventional supervised learning. In weak supervision, instead of experts manually labelling data, samples are given noisy pseudo-labels by using heuristic labelling functions such as domain-specific rules, non-expert annotations, dictionaries, or pattern matching [45]. The pseudo-labels are used to train a machine learning model that can generalise beyond the labelling functions to reach a higher performance. Using weak supervision has been shown to provide performances comparable to that of models trained only on manually labelled data [124, 125].

In the rest of the chapter, I present our work on a weakly supervised model for automatic de-identification of names, streets, and locations in Danish unstructured EHR text.

7.1 Weakly Supervised Model for De-identification of Unstructured EHR Text

This section presents a deep learning model for automatic de-identification of names, streets, and locations in Danish unstructured EHR text using weak supervision. The work on the model is detailed in the following paper:

[49] Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Vinholt, and Thiusius R Savarimuthu. "Automatic Annotation of Training Data for Deep Learning Based De-identification of Narrative Clinical Text". In: *WNLPe-Health 2022: Proceedings of The First Workshop on Context-aware NLP in eHealth (WNLPe-Health 2022)*. CEUR Workshop Proceedings. 2023, pp. 30–44

Below, I briefly present the contents of the paper in the context of this dissertation. The paper is included at the end of this section. It is a reprint of our original paper [49], which is licensed under a CC BY 4.0 license. The CC BY 4.0 license can be found at `https://creativecommons.org/licenses/by/4.0/`.

In this paper, we trained a deep learning model with weak supervision by automatically annotating the training data. We used the PURE entity extraction architecture, presented in detail in chapter 6, and trained it with weak labels to extract names; streets; and locations including cities, municipalities, regions, and provinces. We used Da-BERT as the base of PURE. The combination of transfer learning from the pre-trained LLM and using a weak supervision method allowed us to not use any manually annotated data for the training set.

The heuristic labelling function used information from official sources on names, streets, and locations to define identifiers in unstructured EHR text. Ambiguous identifiers were defined by matching identifiers to non-identifiers. Non-identifiers were defined from multiple text sources from the general and clinical domains which did not contain any of the three identifier types. Whether to use an ambiguous identifier as a weak label was decided by a heuristic function that discarded ambiguous identifiers that occurred more in the corpus than expected for an identifier.

The paper contributes to this dissertation with a trained model based on the PURE entity extraction architecture that de-identifies names, streets, and locations in unstructured EHR text with a sensitivity of 93.43%, precision of 86.10%, and F1 of 89.62%. It correctly de-identifies 91% of locations, and 94% of names and streets.

Automatic Annotation of Training Data for Deep Learning Based De-identification of Narrative Clinical Text

Martin Sundahl Laursen[1], Jannik Skyttegaard Pedersen[1], Pernille Just Vinholt[2] and Thiusius Rajeeth Savarimuthu[1]

[1]The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, Denmark
[2]Department of Clinical Biochemistry, Odense University Hospital, Denmark

Abstract
Electronic health records contain information about patients' medical history which is important for research but the text must be de-identified before use. This study utilized dictionaries constructed from publicly available lists of identifiers to automatically annotate a training dataset for a named entity recognition model to de-identify names, streets, and locations in Danish narrative clinical text. Ambiguous identifiers were not annotated if they occurred more than expected for an identifier. The model had recall 93.43%, precision 86.10%, and F1 89.62%. We found that the model generalized from the training data to achieve better performance than simply using the dictionaries to directly annotate text.

Keywords
de-identification, electronic health records, named entity recognition, automatic annotation, deep learning

1. Introduction

Electronic health records (EHR) contain information about patients' contact with the healthcare system including important information about medical history, e.g. symptoms, diagnoses, and treatments. Diagnoses are also registered using International Classification of Diseases 10 codes for administrative purposes. However, not all relevant patient information is represented in codes, e.g symptoms. Further, codes are often incorrect [1, 2, 3, 4, 5] and can therefore not replace the narrative clinical text in EHRs as a source of information.

Apart from treatment of patients, the EHR data are important for e.g. research and education but as they contain personally identifiable information, explicit consent from the affected individual must be given, or the data must be de-identified before being used for secondary purposes [6, 7]. The US Health Insurance Portability and Accountability Act (HIPAA) defines which identifiers must be removed according to the Safe Harbor method for de-identification[1].

WNLPe-Health 2022, December 18, 2022, Delhi, India
msla@mmmi.sdu.dk (M. S. Laursen); jasp@mmmi.sdu.dk (J. S. Pedersen); pernille.vinholt@rsyd.dk (P. J. Vinholt); trs@mmmi.sdu.dk (T. R. Savarimuthu)
0000-0001-5684-1325 (M. S. Laursen); 0000-0002-7066-1563 (J. S. Pedersen); 0000-0002-2035-0169 (P. J. Vinholt); 0000-0002-2478-8694 (T. R. Savarimuthu)

CEUR Workshop Proceedings (CEUR-WS.org)

[1]Guide available at https://www.hhs.gov/hipaa/for-professionals/privacy/special-topics/de-identification/index.html

The identifiers include, among others, names, street addresses, and locations including city, county, and precinct.

Manual de-identification is a time consuming task and, therefore, large datasets are impractical and expensive to de-identify manually. Natural language processing techniques for automatic de-identification may alleviate this task.

This study utilizes dictionaries of identifiers and a novel way of dealing with ambiguous identifiers to automatically annotate a training dataset for a named entity recognition (NER) model to de-identify names, streets, and locations in Danish narrative clinical text.

A method for automatic annotation of training datasets is useful for developing de-identification deep learning models for low-resource languages like Danish where annotated datasets and trained models for de-identification of specific identifier types are not always publicly available.

The main contributions of this paper are:

- We train a NER model to de-identify names, streets, and locations in Danish narrative clinical text with recall 93.43%, precision 86.10%, and F1 89.62%.
- We use dictionary-based automatic annotation of training data for the NER model utilizing our novel method for annotation of ambiguous identifiers guided by occurrence rates in the text and population.
- We find that the NER model can generalize from the dataset to achieve better performance than simply using the dictionaries to directly annotate text.

2. Related Work

Previous studies on automatic de-identification of narrative clinical text used rule-based methods, machine learning methods, and hybrid methods combining both. We found no studies that, similar to ours, used automatic annotation of data for training a machine learning model to de-identify narrative clinical text.

In studies that used rule-based methods, pattern matching or dictionaries were used to search for identifiers in the text [8, 9, 10, 11, 12, 13]. Rule-based methods rely on domain experts to define the rules and it is difficult to cover all cases. They generally cannot distinguish ambiguous identifiers, i.e. words that can both be an identifier and a non-identifier depending on the context. Pantazos et al. [12] de-identified Danish text using dictionaries with an F1 score of 95.7% on a random sample of 369 EHRs. They identified ambiguous identifiers by matching identifiers to a database of non-identifiers. As they replaced identifiers with pseudo-identifiers, their approach to ambiguous identifiers was to delete the record unless the identifier appeared more than 200 times to not disclose their replacement rule.

Studies that used machine learning methods mainly used recurrent neural networks, conditional random fields, and combinations of the two [14, 15, 16, 17, 18, 19]. All studies that utilized machine learning used a manually annotated dataset for training the models. Machine learning methods and in particular deep learning architectures such as Long Short-Term Memory [20] and transformer [21] networks are able to distinguish ambiguous identifiers based on the context of the whole sentence. Some recent studies have used transformer networks for automatic de-identification of narrative clinical text [22, 19, 23, 24, 25, 26]. A disadvantage of machine learning methods is their need for a large expert-annotated training dataset specific to the domain.

Finally, the studies that were most similar to ours used hybrid methods, combining rule-based and machine learning methods in ensembles or pipelines to improve the annotation workload and model performance [27, 28, 29, 30, 25]. Two studies used rule-based methods in other ways than for directly classifying identifiers. McMurry et al. [27] used pattern matching to contribute part of a feature set for classification by a machine learning model which resulted in a F1 score of 76% on a custom test set of 220 discharge summaries. Jian et al. [28] used pattern matching to create a dense corpus of identifiers for manual annotation before being input to a machine learning model. It had an F1 score of 94.6% when cross-validating on 3,000 clinical documents.

3. Methods

In this paper, we first constructed lists of name, street, and location identifiers. We compared the identifiers to a database of non-identifying words to determine which identifiers were ambiguous—e.g. the name 'Hans' is ambiguous because it is also a pronoun (Danish for 'his'). This dictionary-based method is similar to that of e.g. Pantazos et al. [12] except in this paper, we used it to annotate training data for a deep learning model instead of using direct dictionary-based de-identification. Additionally, we utilized a novel method for annotation of ambiguous identifiers and tested different ceiling values above which words were removed from the list of identifiers if they occured in the text at a higher rate than would be expected for an identifier. We searched for and annotated identifiers in Danish narrative clinical text and constructed a training set of sentences with no or only unambiguous identifiers. Finally, the training set was used to train a NER de-identification model. The goal was for the model to generalize from the training samples with no or only unambiguous identifiers to also correctly classify ambiguous identifiers. This process is detailed in the rest of this section. We make our code publicly available[2].

3.1. Data

3.1.1. Corpus

We extracted 150,000 random sentences with a length between 8 and 70 words from EHRs from Odense University Hospital between 2015 and 2020. Sentences were lowercased and tokenized, and consecutive underscores and hyphens were reduced to a single instance.

3.1.2. Identifiers

The identifier types were names, streets, and locations. Locations included cities, municipalities, regions, and provinces.

For the name identifiers, we obtained lists of all male first names, female first names, and last names in the Danish population as of January 2021 from Statistics Denmark.

[2] https://github.com/jannikskytt/clinical_de-identification

For the street identifiers, we used a database of all Danish addresses from the Address Web Services of the Agency for Data Supply and Efficiency of Denmark[3]. Each address included street name, addressing street name (could be identical to street name), city name, potential supplemental city name, municipality, region, and province. From the database, a list of unique street names including addressing street names was constructed.

For the location identifiers, we used the same database of all Danish addresses. A list of unique locations including city names, supplemental city names, municipalities, regions, and provinces was constructed.

Data cleaning consisted of lowercasing and removing single-letter and empty and corrupted identifiers including various placeholders.

A rate of occurrence in the Danish population was calculated for each identifier by dividing their occurrence in the population by the sum of all occurrences for that identifier type. For each identifier type, duplicates were merged by adding the rates of occurrence.

3.1.3. Non-identifiers

Non-identifiers were words that in their context did not identify names, streets, or locations. Such words included both common general domain words and specialized words from the clinical domain such as symptoms, diseases, and treatments. The database of non-identifiers was constructed from multiple text sources from the general and clinical domains which did not contain any of the three identifier types.

The text sources were:

- The Danish orthographic dictionary containing all Danish words, their conjugations, and abbreviations [31].
- Product names from the list of authorized medicinal products in Denmark[4].
- Medical abbreviations collected from different electronic sources (Appendix A)
- All term entries in the Description tables of the SNOMED CT vocabulary of clinical terminology (international version with Danish extension).
- The Danish healthcare system's classification system for symptoms, diagnoses, and operations[5].

3.2. Ambiguous Identifiers

An identifier could be ambiguous for two reasons. One reason was that it had multiple different identifier types, e.g. 'Kolding' is both a location and a name. In that case, the identifiers' rates of occurrence were added. Another reason was that it was also a non-identifier. To find those cases, identifiers were matched against the database of non-identifiers using a regular expression that ignored case (regex). If an identifier was matched to a non-identifier, it was ambiguous.

[3]All datasets were downloaded from https://download.aws.dk/
[4]Available at https://laegemiddelstyrelsen.dk/en/
[5]Available at https://sundhedsdatastyrelsen.dk/

3.3. Automatic Annotation

For the automatic annotation of identifiers in sentences, specifically dealing with ambiguous identifiers, we introduced a measure for the likelihood of a word being a non-identifier vs. identifier for the specific corpus. The measure was the ratio between the rate with which the word occurred in sentences in the corpus as either identifier or non-identifier, and the rate of occurrence in the Danish population as identifier: $ratio = r_{corpus}/r_{population}$. A ratio above 1 meant that the word had a higher rate of occurrence in the corpus than as an identifier in the Danish population. This could indicate that it in most cases occurred as a non-identifier in the corpus. A ratio below 1 could indicate that the word in most cases occurred as an identifier.

The rate of occurrence in the corpus was calculated for each identifier by searching through all sentences using a regex, counting the number of occurrences, and dividing by the total number of sentences. The ratio was then calculated using the equation.

Next, a regex was used to search for and annotate identifiers in the sentences. Words that were unambiguous identifiers were annotated with their single identifier type. Words that were ambiguous because they had multiple identifier types were annotated with both. Words that were ambiguous because they were both an identifier and a non-identifier were annotated with their identifier type and a non-identifier tag with two exceptions: (1) if the ratio was below 1, they were annotated only with their identifier type, and (2) if the ratio was above a set ratio ceiling, the identifier was not annotated, i.e. kept as a non-identifying word.

Finally, all annotated sentences were postprocessed in the following order:

1. If an ambiguous identifier was the same type as a neighbor identifier, it was converted to that type.
2. If a single letter was between two name identifiers, it was taken as a middle initial and converted to a name identifier.
3. Identifiers of the same type which were next to each other were converted to a single identifier consisting of multiple words.

We tested values for the ratio ceiling on a binary logarithmic scale from 1 to 262,144.

3.4. Named Entity Recognition Model

We used the automatically annotated sentences to create multiple datasets, based on different values for the ratio ceiling, for training Princeton University Relation Extraction system (PURE) [32] NER models to de-identify name, street, and location identifiers in the corpus.

3.4.1. Datasets

The validation and test sets each contained 1,500 sentences. They were annotated for names, streets, and locations by one of the authors using the CLAMP software [33]. The sentences for the validation and test sets were selected by setting the ratio ceiling to the median ratio of all identifiers and choosing 500 sentences with no identifiers, 500 with only unambiguous identifiers, and 500 with at least one ambiguous identifier. The distributions of types of ambiguous and unambiguous identifiers were approximately the same as in the entire corpus. Selecting the validation and test sets in this way ensured that as many models as possible would experience varying sentences with regards to types, ambiguity, and number of identifiers.

While the validation and test sets were human annotated and fixed for all models, the training sets were annotated automatically using the described method and varied with each of the tested ratio ceilings used for the automatic annotation. Training sets were constructed from all sentences not used for the validation and test sets. Only sentences with no or unambiguous identifiers were selected for the training sets since the NER model was only trained with unambiguous samples. In cases where the number of sentences containing no identifiers was higher than the number containing identifiers, the former was downsampled to the latter.

All datasets were converted to the structure used by PURE.

3.4.2. Training of Model

For each training set automatically annotated with the different ratio ceilings, a PURE NER model was trained with a publicly available uncased Danish pretrained BERT [34] model[6] as base. The default hyperparameters of PURE were used (see [32]) except a context window of 0. Models were trained until convergence (maximum 100 epochs). The F1 score on the validation set was used to select the best model checkpoint from each training.

3.4.3. Evaluation of Model

The best performing model on the current data was found by evaluating the F1 scores on the test set. Performance on the three identifier types was evaluated in a confusion matrix. Additionally, for each model, we compared its test set performance to that of the dictionary-based method used for annotating its training set to see if the model generalized from its training data to improve performance.

The ratio ceiling used for automatic annotation of the training set for the best performing model was tested for model training with less available data to evaluate the minimum amount needed for top model performance.

Finally, we analyzed the effect of lowering the ratio ceiling to produce more training samples when there was less data than the minimum amount needed for top model performance.

4. Results

4.1. Identifiers

The list of identifiers had 449,997 unambiguous identifiers: 397,348 names, 48,859 streets, and 3,790 locations. 18,057 identifiers were ambiguous: 16,582 had a name type, 2,505 a street type, and 3,133 a location type. 3,859 of the ambiguous identifiers had more than one identifier type. 7,148 ambiguous identifiers matched a non-identifier in the Danish orthographic dictionary, 312 in authorized medicinal products, 406 in medical abbreviations, 9,890 in SNOMED CT, and 2,013 in the healthcare system's classification system. Identifiers had rates of occurrence in the population between 8.58e-06% and 30.39% with median 1.72e-05%.

[6] Available at https://github.com/certainlyio/nordic_bert

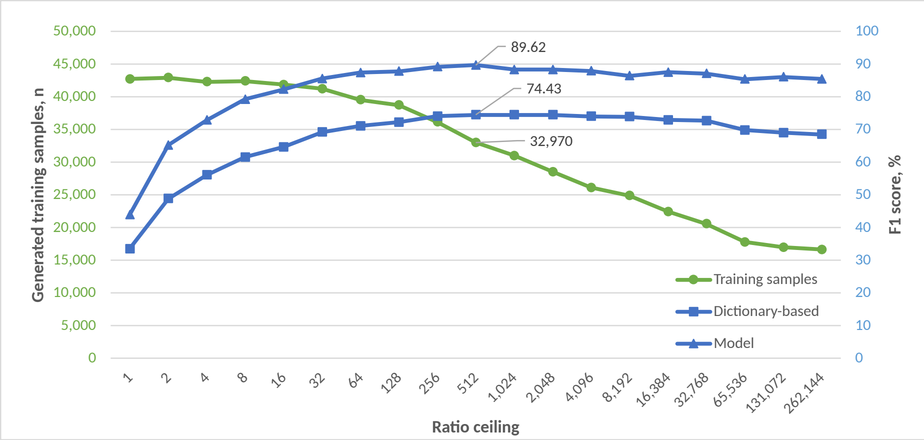

Figure 1: Model F1 and dictionary-based F1 in blue, right axis. Amount of samples in the training set in green, left axis.

Automatically annotated identifiers had corpus vs. population ratios between 4.42e-03 and 9.82e+06 (median 306.73). The highest ratio was 'københavn' (ambiguous location) while the lowest ratio was 'og' (ambiguous name and conjunction 'and').

4.2. Named Entity Recognition Model

Table 1
Identifiers in the validation and test sets.

	Validation set	Test set
Name	943	965
Street	77	97
Location	353	324
Total	1,373	1,386

Table 1 shows the distribution of identifier types in the human annotated validation and test sets.

Figure 1 shows the test set F1 scores for the models which training sets were automatically annotated with different ratio ceilings. The F1 score of the dictionary-based method and the amount of samples in the training sets are also plotted (further details in Appendix B).

The best model was trained with data automatically annotated with a 512 ratio ceiling (32,970 training set samples). It had a recall of 93.43%, a precision of 86.10% and a F1 of 89.62%. There was an upwards trend in the 1–512 ratio ceilings and downwards in 512–262,144. All model F1 scores were higher than the corresponding dictionary-based F1 scores. Most training set samples were produced with the ratio ceiling at 2 (42,894) while the least were produced by ratio ceiling 262,144 (16,614). Training time for the best model was 2 hours 4 minutes for 20

Figure 2: Test set confusion matrix. 'O' is non-identifiers for which only errors were counted.

epochs on a Nvidia Tesla v100 GPU.

Figure 2 shows the confusion matrix for model performance. 94% of street and name identifiers, and 91% of location identifiers were classified correctly. Non-identifiers were most often misclassified as names (75% of misclassifications).

Comparing test set performance to the dictionary-based method, the model correctly classified 283 identifiers that the dictionary-based method misclassified. The dictionary-based method correctly classified 13 identifiers that the model misclassified. 70 identifiers were misclassified by both the model and the dictionary-based method. Appendix C shows the performance of the model and the dictionary-based method on words that occurred in the test set both as non-identifiers and identifiers. E.g., for the word 'per', the model correctly classified it as an identifier (name) in 100% of cases and as a non-identifier (preposition: 'per') in 91% of cases. For the dictionary-based method, it was 57% and 100%, respectively. Note that the dictionary-based method could classify the same word differently because of the postprocessing steps where an ambiguous identifier could be converted to an unambiguous identifier under certain conditions. Among all words that occurred both as non-identifiers and identifiers, the model classified 92% of non-identifiers and 84% of identifiers correctly. For the dictionary-based method, it was 96% and 50%, respectively.

4.3. Analysis of Ratio Ceiling

We analyzed the effect of lowering the ratio ceiling to produce more training samples when there was less data than needed for top model performance.

The best performing model was trained on data automatically annotated with ratio ceiling 512 and had 147,000 sentences available from which 32,970 sentences were used for the training set. We lowered the amount of available data for automatic annotation with ratio ceiling 512 from 147,000 through to 12,000 sentences without any reduction in performance.

Next, we tested the effects on which ratio ceiling was the best when lowering the amount of available data below 12,000 sentences. We included ratio ceilings between 512 and 16 since they

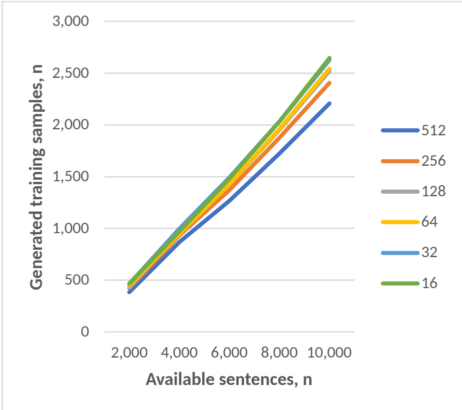

Figure 3: Amount of training set samples generated from the available data by automatically annotating with ratio ceilings 16–512.

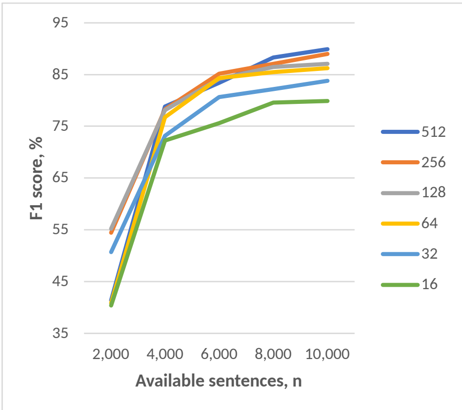

Figure 4: F1 scores for models with training sets automatically annotated with ratio ceilings 16–512 by amount of available data.

generated increasingly more samples for the training set (Figure 3). When the available data was less than 8,000 sentences, performance with lower ratio ceilings surpassed that of the 512 ratio ceiling in some cases (Figure 4).

5. Discussion

We used an automatically annotated training set to train a PURE NER deep learning model to de-identify names, streets and locations in Danish narrative clinical text with a recall of 93.43%, precision of 86.10%, and F1 score of 89.62%. Non-identifiers were most often misclassified as names which may be caused by greater variability than for streets and locations.

We took a similar approach as Pantazos et al. [12] to identify ambiguous identifiers through matching of identifiers to a database of non-identifiers. While, for de-identification, they deleted records of ambiguous identifiers that occurred less than 200 times, we trained a deep learning model from an automatically annotated training set to de-identify ambiguous identifiers. For the automatic annotation, we handled ambiguous identifiers by calculating the ratio between the rate of occurrence in the corpus and the rate of occurrence in the population for every identifier. This method allowed an individual assessment if they should be annotated as an identifier or not in the training data—increasing the chance of model generalization. The ratio ceiling also allowed to balance the quality and amount of training data. Analyzing the ratio ceiling, we found that when less than 8,000 sentences were available, the extra samples provided by a lower ratio ceiling became more important than using the ratio ceiling that gave the highest quality of the training data. Lower ratio ceilings produced more training data because more ambiguous identifiers were considered non-identifiers resulting in fewer ambiguous sentences that had to be discarded from the training set.

We saw an increase in F1 from dictionary-based de-identification to annotating a training set with the dictionary-based method, training a NER model, and de-identifying with the trained model. This showed that the model generalized from the training data to better classify the ambiguous identifiers that the dictionary-based approach could not differentiate, and achieve better performance than simply using the dictionaries to directly annotate text. This is supported by the model correctly de-identifying 84% of words that occurred in the test set both as identifier and non-identifier. Only 50% of these words were de-identified by the dictionary-based method.

5.1. Limitations

It is a limitation to the study that the data came only from Odense University Hospital but the ratios were calculated using the rate of occurrence in the entire population of Denmark.

Future work includes de-identification of the rest of the HIPAA Safe Harbor identifiers since there is no guarantee that the presented methods will generalize to other identifiers. Since this study used lowercased data because only a lowercased Danish BERT base was available, exploring performance when keeping the case of training data is also part of future work.

6. Conclusions

We trained a NER deep learning model using automatically annotated data to de-identify names, streets, and locations in Danish narrative clinical text with recall 93.43%, precision 86.10%, and F1 89.62%. A model trained on data annotated with a dictionary-based method can generalize and surpass the performance of the dictionary-based method. A ratio ceiling of 512 works best for Danish narrative clinical text when more than 8,000 sentences are available.

The automatic de-identification method presented in this study can be adapted to all languages and domains if lists of identifiers and non-identifiers are available. Apart from the lists, the method does not need any external data as the input data to the de-identification model is used to train the model itself. This makes the method particularly useful for low-resource languages where annotated datasets and trained models for de-identification of specific identifier types are not always publicly available.

References

[1] V. E. Valkhoff, P. M. Coloma, G. M. Masclee, R. Gini, F. Innocenti, F. Lapi, M. Molokhia, M. Mosseveld, M. S. Nielsson, M. Schuemie, F. Thiessard, J. van der Lei, M. C. Sturkenboom, G. Trifirò, Validation study in four health-care databases: upper gastrointestinal bleeding misclassification affects precision but not magnitude of drug-related upper gastrointestinal bleeding risk, Journal of Clinical Epidemiology 67 (2014) 921–931. URL: https://www.sciencedirect.com/science/article/pii/S0895435614000845. doi:https://doi.org/10.1016/j.jclinepi.2014.02.020.

[2] L. R. Øie, M. A. Madsbu, C. Giannadakis, A. Vorhaug, H. Jensberg, Ø. Salvesen, S. Gulati, Validation of intracranial hemorrhage in the norwegian patient registry, Brain and Behavior 8 (2018) e00900. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/brb3.900. doi:https://doi.org/10.1002/brb3.900. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/brb3.900.

[3] J. Delekta, S. M. Hansen, K. S. AlZuhairi, C. S. Bork, A. M. Joensen, The validity of the diagnosis of heart failure (I50.0-I50.9) in the danish national patient register, Dan Med J 65 (2018).

[4] T. L. Higgins, A. Deshpande, M. D. Zilberberg, P. K. Lindenauer, P. B. Imrey, P.-C. Yu, S. D. Haessler, S. S. Richter, M. B. Rothberg, Assessment of the Accuracy of Using ICD-9 Diagnosis Codes to Identify Pneumonia Etiology in Patients Hospitalized With Pneumonia, JAMA Network Open 3 (2020) e207750–e207750. URL: https://doi.org/10.1001/jamanetworkopen.2020.7750. doi:10.1001/jamanetworkopen.2020.7750. arXiv:https://jamanetwork.com/journals/jamanetworkopen/articlepdf/2768537/higgin

[5] N. Wabe, L. Li, R. Lindeman, J. J. Post, M. R. Dahm, J. Li, J. I. Westbrook, A. Georgiou, Evaluation of the accuracy of diagnostic coding for influenza compared to laboratory results: the availability of test results before hospital discharge facilitates improved coding accuracy, BMC Medical Informatics and Decision Making 21 (2021) 168. URL: https://doi.org/10.1186/s12911-021-01531-9. doi:10.1186/s12911-021-01531-9.

[6] GDPR, Regulation (eu) 2016/679 (general data protection regulation), ???? URL: https://gdpr-info.eu/.

[7] HIPAA, Health insurance portability and accountability act of 1996 (hipaa), public law 104-191, ???? URL: https://www.hhs.gov/hipaa/for-professionals/index.html.

[8] B. A. Beckwith, R. Mahaadevan, U. J. Balis, F. Kuo, Development and evaluation of an open source software tool for deidentification of pathology reports, BMC Medical Informatics and Decision Making 6 (2006) 12.

[9] I. Neamatullah, M. M. Douglass, L.-W. H. Lehman, A. Reisner, M. Villarroel, W. J. Long, P. Szolovits, G. B. Moody, R. G. Mark, G. D. Clifford, Automated de-identification of free-text medical records, BMC Medical Informatics and Decision Making 8 (2008) 32.

[10] E. Chazard, C. Mouret, G. Ficheur, A. Schaffar, J.-B. Beuscart, R. Beuscart, Proposal and evaluation of fasdim, a fast and simple de-identification method for unstructured free-

text clinical records, International Journal of Medical Informatics 83 (2014) 303–312. URL: https://www.sciencedirect.com/science/article/pii/S1386505613002463. doi:https://doi.org/10.1016/j.ijmedinf.2013.11.005.

[11] S. Y. C. H. J. P. J. L. Y. L. M.-S. C. C.-M. K. W.-S. L. J. H. Shin Soo-Yong, Park Yu Rang, A de-identification method for bilingual clinical texts of various note types, jkms 30 (2015) 7–15. URL: http://www.e-sciencecentral.org/articles/?scid=1022920. doi:10.3346/jkms.2015.30.1.7. arXiv:http://www.e-sciencecentral.org/articles/?scid=1022920.

[12] K. Pantazos, S. Lauesen, S. Lippert, Preserving medical correctness, readability and consistency in de-identified health records, Health Informatics Journal 23 (2017) 291–303. URL: https://doi.org/10.1177/1460458216647760. doi:10.1177/1460458216647760. arXiv:https://doi.org/10.1177/1460458216647760.

[13] V. Menger, F. Scheepers, L. M. van Wijk, M. Spruit, Deduce: A pattern matching method for automatic de-identification of dutch medical text, Telematics and Informatics 35 (2018) 727–736. URL: https://www.sciencedirect.com/science/article/pii/S0736585316307365. doi:https://doi.org/10.1016/j.tele.2017.08.002.

[14] H. Fabregat, A. Duque, J. Martinez-Romo, L. Araujo, De-identification through named entity recognition for medical document anonymization, Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019) (2019).

[15] K. Kajiyama, H. Horiguchi, T. Okumura, M. Morita, Y. Kano, De-identifying free text of japanese electronic health records, Journal of Biomedical Semantics 11 (2020) 11.

[16] L. Lange, H. Adel, J. Strötgen, Closing the gap: Joint de-identification and concept extraction in the clinical domain, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 6945–6952. URL: https://aclanthology.org/2020.acl-main.621. doi:10.18653/v1/2020.acl-main.621.

[17] J. L. Leevy, T. M. Khoshgoftaar, F. Villanustre, Survey on RNN and CRF models for de-identification of medical free text, Journal of Big Data 7 (2020) 73.

[18] I. Pérez-Díez, R. Pérez-Moraga, A. López-Cerdán, J.-M. Salinas-Serrano, M. d. la Iglesia-Vayá, De-identifying spanish medical texts - named entity recognition applied to radiology reports, Journal of Biomedical Semantics 12 (2021) 6.

[19] R. Catelli, V. Casola, G. De Pietro, H. Fujita, M. Esposito, Combining contextualized word representation and sub-document level analysis through bi-lstm+crf architecture for clinical de-identification, Knowledge-Based Systems 213 (2021) 106649. URL: https://www.sciencedirect.com/science/article/pii/S0950705120307784. doi:https://doi.org/10.1016/j.knosys.2020.106649.

[20] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (1997) 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[22] R. Catelli, F. Gargiulo, V. Casola, G. De Pietro, H. Fujita, M. Esposito, Crosslingual named entity recognition for clinical de-identification applied to a covid-19 italian data set, Applied

Soft Computing 97 (2020) 106779. URL: https://www.sciencedirect.com/science/article/pii/S1568494620307171. doi:https://doi.org/10.1016/j.asoc.2020.106779.

[23] R. Catelli, F. Gargiulo, E. Damiano, M. Esposito, G. De Pietro, Clinical de-identification using sub-document analysis and electra, in: 2021 IEEE International Conference on Digital Health (ICDH), 2021, pp. 266–275. doi:10.1109/ICDH52753.2021.00050.

[24] R. Catelli, F. Gargiulo, V. Casola, G. De Pietro, H. Fujita, M. Esposito, A novel covid-19 data set and an effective deep learning approach for the de-identification of italian medical records, IEEE Access 9 (2021) 19097–19110. doi:10.1109/ACCESS.2021.3054479.

[25] K. Murugadoss, A. Rajasekharan, B. Malin, V. Agarwal, S. Bade, J. R. Anderson, J. L. Ross, W. A. Faubion, J. D. Halamka, V. Soundararajan, S. Ardhanari, Building a best-in-class automated de-identification tool for electronic health records through ensemble learning, Patterns 2 (2021) 100255. URL: https://www.sciencedirect.com/science/article/pii/S2666389921000817. doi:https://doi.org/10.1016/j.patter.2021.100255.

[26] C. Meaney, W. Hakimpour, S. Kalia, R. Moineddin, A comparative evaluation of transformer models for de-identification of clinical text data, 2022. URL: https://arxiv.org/abs/2204.07056. doi:10.48550/ARXIV.2204.07056.

[27] A. J. McMurry, B. Fitch, G. Savova, I. S. Kohane, B. Y. Reis, Improved de-identification of physician notes through integrative modeling of both public and private medical text, BMC Medical Informatics and Decision Making 13 (2013) 112.

[28] Z. Jian, X. Guo, S. Liu, H. Ma, S. Zhang, R. Zhang, J. Lei, A cascaded approach for chinese clinical text de-identification with less annotation effort, Journal of Biomedical Informatics 73 (2017) 76–83. URL: https://www.sciencedirect.com/science/article/pii/S1532046417301776. doi:https://doi.org/10.1016/j.jbi.2017.07.017.

[29] Y. Kim, P. Heider, S. Meystre, Ensemble-based methods to improve de-identification of electronic health record narratives, AMIA Annu Symp Proc 2018 (2018) 663–672.

[30] P. Richter-Pechanski, S. Riezler, C. Dieterich, De-Identification of german medical admission notes, Stud Health Technol Inform 253 (2018) 165–169.

[31] Danish Language Council, Retskrivningsordbogen, 4th edition, 2017, including 8 digital issues, Danish Language Council, Bogense, Denmark, 2012.

[32] Z. Zhong, D. Chen, A frustratingly easy approach for entity and relation extraction, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online, 2021, pp. 50–61. URL: https://aclanthology.org/2021.naacl-main.5. doi:10.18653/v1/2021.naacl-main.5.

[33] E. Soysal, J. Wang, M. Jiang, Y. Wu, S. Pakhomov, H. Liu, H. Xu, CLAMP – a toolkit for efficiently building customized clinical natural language processing pipelines, Journal of the American Medical Informatics Association 25 (2017) 331–336. URL: https://doi.org/10.1093/jamia/ocx132. doi:10.1093/jamia/ocx132. arXiv:https://academic.oup.com/jamia/article-pdf/25/3/331/34150625/ocx132.pdf.

[34] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational

Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

A. Medical Abbreviations

Table 2
Sources of medical abbreviations

Description	Link	Deep link
List of recognized abbreviations and symbols used in Hospital Sønderjylland	skoletube.dk	https://www.skoletube.dk/download.php?key=MGVhNTVjOTFkZDdiYjIyNDU0ZGI
Symbols and abbreviations used at the laboratory, Hospital Sønderjylland	regionsyddanmark.dk	https://shs.regionsyddanmark.dk/wm272641
Abbreviations and symbols in patient related data	regionsjaellend.dk	http://dok.regionsjaelland.dk/view.aspx?DokID=217238
Definitions and abbreviations, Department of clinical biochemistry, Herlev and Gentofte Hospital	gentoftehospital.dk	https://www.gentoftehospital.dk/afdelinger-og-klinikker/klinisk-biokemisk-afdeling/metodeblade/Documents/Definitioner%20og%20forkortelser%20%E2%80%93%20Klinisk%20Biokemisk%20Afdeling_2020_11_03.pdf
Abbreviations and symbols, psychiatry in Region Nordjylland	pri.rn.dk	https://pri.rn.dk/Assets/15764/Forkortelser-og-symboler-gaeldende-for-sygeplejefaglige-optegnelser.pdf
Medical abbreviations	medicinskefagudtryk.dk	http://medicinskefagudtryk.dk/sidebar/forkortelser/
Ordinary medical abbreviations	medviden.dk	https://www.medviden.dk/vaerktoejer/andre/almindelige-medicinske-forkortelser/
Guidelines for journal writing	gyldendal.dk	http://guga.gyldendal.dk/Munksgaard/$\sim$/media/Munksgaard/Medicinsk%20Fagsprog%202/Retningslinjer.ashx
Abbreviations and designations	medicin.dk	https://pro.medicin.dk/Artikler/Artikel/215
Abbreviations used in Department R, FMB	docplayer.dk	https://docplayer.dk/5008471-Forkortelser-anvendt-i-afd-r-s-instrukser-og-vejledninger-120204-fmb.html
Clinical biochemistry test results	pri.rn.dk	https://pri.rn.dk/Sider/10307.aspx
Family tree abbreviations	pri.rn.dk	https://pri.rn.dk/Sider/29580.aspx
Abbreviations and symbols in the eye speciality	pri.rn.dk	https://pri.rn.dk/Sider/17827.aspx
Accepted abbreviations at vascular surgery department	pri.rn.dk	https://pri.rn.dk/Sider/17293.aspx
Renal abbreviations	pri.rn.dk	https://pri.rn.dk/Sider/16695.aspx
Heart Lung surgery abbreviations	pri.rn.dk	https://pri.rn.dk/Sider/10530.aspx
Intensive Therapy abbreviations	docplayer.dk	https://docplayer.dk/11055021-Hyppigt-anvendte-forkortelser-og-termer-i-intensiv-terapi.html

B. Results Table

Table 3
Distribution of identifiers in each of the training sets automatically annotated with different ratio ceilings and the resulting model and dictionary-based F1 scores.

Ratio ceiling	Sentences with identifiers	Sentences without identifiers	Total sentences	Dictionary-based F1 %	Model F1 %	Name tags	Street tags	Location tags	Total tags
1	21,343	21,343	42,686	33.45	43.86	26,775	1,015	4,309	32,099
2	21,447	21,447	42,894	48.80	65.17	29,003	1,128	3,870	34,001
4	21,131	21,131	42,262	56.02	72.79	28,596	1,217	3,781	33,594
8	21,185	21,185	42,370	61.52	79.19	28,388	1,437	3,749	33,574
16	20,923	20,923	41,846	64.56	82.21	27,988	1,508	3,803	33,299
32	20,597	20,597	41,194	69.11	85.52	27,342	1,660	3,707	32,709
64	19,750	19,750	39,500	71.05	87.36	26,382	2,130	3,533	32,045
128	19,354	19,354	38,708	72.19	87.73	25,963	2,248	3,446	31,657
256	18,058	18,058	36,116	74.02	89.10	24,191	2,391	3,138	29,720
512	16,485	16,485	32,970	74.43	89.62	22,277	2,396	2,897	27,570
1,024	15,490	15,490	30,980	74.42	88.24	21,001	2,323	2,652	25,976
2,048	14,246	14,246	28,492	74.42	88.25	19,617	2,290	2,475	24,382
4,096	13,041	13,041	26,082	73.97	87.85	18,127	2,205	2,332	22,664
8,192	12,424	12,424	24,848	73.82	86.35	17,450	2,196	2,244	21,890
16,384	11,205	11,205	22,410	72.84	87.48	15,826	2,133	1,931	19,890
32,768	10,278	10,278	20,556	72.63	87.06	14,732	2,090	1,756	18,578
65,536	8,879	8,879	17,758	69.77	85.34	12,848	1,915	1,462	16,225
131,072	8,482	8,482	16,964	68.98	86.01	12,330	1,903	1,380	15,613
262,144	8,307	8,307	16,614	68.46	85.37	12,295	1,893	1,291	15,479

C. Ambiguous Performance

Table 4
A comparison of the model and dictionary-based performance on words that occurred both as non-identifiers and identifiers in the test set. Only rows where there is a difference in performance are shown. The total is calculated over all words.

	Non-identifiers		Identifiers	
	Dictionary-based % (total)	**Model % (total)**	**Dictionary-based % (total)**	**Model % (total)**
per	100% (32)	91% (32)	57% (7)	100% (7)
hans	100% (25)	88% (25)	20% (5)	100% (5)
maria	100% (4)	75% (4)	38% (13)	100% (13)
ringe	100% (11)	100% (11)	0% (3)	100% (3)
plads	100% (9)	100% (9)	0% (1)	100% (1)
rask	100% (8)	88% (8)	50% (2)	100% (2)
bak	100% (1)	0% (1)	100% (4)	100% (4)
bo	100% (1)	100% (1)	50% (4)	100% (4)
do	100% (3)	100% (3)	100% (1)	0% (1)
tønder	100% (1)	100% (1)	0% (3)	100% (3)
hammer	100% (1)	0% (1)	100% (2)	100% (2)
rene	100% (1)	100% (1)	50% (2)	100% (2)
slagelse	100% (2)	0% (2)	0% (1)	100% (1)
hammel	100% (1)	100% (1)	0% (1)	100% (1)
land	100% (1)	100% (1)	0% (1)	100% (1)
langeland	100% (1)	0% (1)	0% (1)	100% (1)
stokke	100% (1)	100% (1)	0% (1)	100% (1)
Total	**96% (296)**	**92% (296)**	**50% (88)**	**84% (88)**

7.2 Concluding Remarks

In this chapter, I presented the work of a paper that used weak supervision to train a PURE entity extraction model to de-identify names, streets, and locations in unstructured EHR text. It performs with a sensitivity of 93.43%, precision of 86.10%, and F1 of 89.62% and correctly de-identifies 91% of locations, and 94% of names and streets.

The Data Minimisation Principle

The data minimisation principle of the GDPR states that processed personal data must be limited to what is necessary in relation to the purposes of processing [40]. The models presented in chapters 4, 5, and 6 can facilitate minimisation by only extracting the information needed for clinical research rather than using all unstructured text. The combination of using the de-identification model presented here and models for classification and extraction of only relevant data complies with two principles of the GDPR.

Limitations and Future Work

Limitations to clinical applicability is that the model only de-identifies names, locations, and streets. The presented method must be applied to more identifiers in the future, or the method can be combined with rule-based de-identification of identifiers such as dates, phone numbers, and e-mail addresses.

Other future work includes testing more semi-supervised methods such as self-training and active learning, and the zero-shot prompt-based method applied by Liu et al. [111].

CHAPTER 8

Faster Annotation of a Dense Corpus

In this chapter, I train Danish clinical word embeddings that can be used to create a dense clinical annotation corpus and increase the efficiency of the annotation process. This addresses research question 5.

All models presented in this dissertation can potentially improve their performance with continuous collection of data, annotation of new samples, and re-training. Semi-supervised methods such as active learning can be utilised for improving existing models with minimal annotation effort. Future work is needed, though, on models for extraction of other parameters for clinical research than those presented in this dissertation. Data collection and annotation of samples for new models to be built from scratch is very time-consuming and expensive. Weak supervision, as discussed in chapter 7, can be applied to some tasks, but other tasks require a manual annotation process for optimal model performance.

In the rest of the chapter, I first describe how a dense annotation corpus can make annotation more efficient, and a method to construct the dense annotation corpus using word embeddings. Then, I present our work on training Danish clinical word embeddings.

8.1 Creating a Dense Annotation Corpus

When annotating for a new model for extraction of binary clinical research parameters, like indication of bleeding, the corpus often contains a much higher proportion of negative samples than positive. If annotating for a multi-class model, e.g. for bleeding location classification, some classes might be much less frequent in the corpus—as was the case for muscle and joint bleedings in chapter 4. Annotating enough data to collect a sufficient amount of positive samples, or a balanced dataset for multi-class models, can be very time-consuming and expensive. Creating a dense annotation corpus with a more equal proportion of positive and negative samples, or between classes, can increase the efficiency of the annotation process for new models.

One approach to creating a dense annotation corpus of EHRs, is sampling them based on ICD codes. If annotating for, e.g. bleeding, the corpus can be filtered to only consist of EHRs that have an ICD code for bleeding. As described in the introduction, though, ICD codes for bleeding are not always accurate, and annotators could have to read through an entire admission to find the sentence that indicates a bleeding.

Another, more fine-grained, approach to creating a dense annotation corpus is based on only including sentences that contain specific words. This can be used to even out the proportion of negative and positive samples or increase the frequency of a specific class. Only the sentences that match the words in a pre-constructed dictionary are included in the corpus for annotation [126]. The dictionary can be created by initially selecting a pool of words indicating the clinical concept that is to be annotated. For bleeding, such words would be "bleeding", "bleeds", "epistaxis", "haematuria", "haemorrhage", etc. As it is a complicated task to detect all relevant words that can describe a clinical concept, word embeddings can be used to expand the list of initial words by iteratively finding the nearest neighbours to each word in the list, and expanding it with those confirmed by a human.

It is important that the word embeddings used to expand the dictionary are suited for the clinical domain and perform optimally. This can be ensured by benchmarking different word embedding models, hyperparameters, and training data.

8.2 Non-contextual Word Embeddings

Non-contextual word embeddings are dense, fixed-length vector representations of words that encode syntactic and semantic word relationships [127]. They are trained on a large corpus in an unsupervised fashion using word co-occurrence statistics. Several different methods of encoding the information exist but all are based on the distributional hypothesis [128]—the assumption that words with similar context have the same meaning [127].

Non-contextual word embeddings differ from contextual LLMs such as BERT and ELECTRA in that the representation does not consider the context of the sequence [78]. They can be used as input features to NLP models and are suited for non-contextual tasks such as extracting the encoded syntactic and semantic information on words, e.g. by identifying a word's nearest neighbours in word embedding space [129]. A limitation is that homographs share one representation that can be considered an average of the two words. One example is "hunter", as in "hunter syndrome", and a person who hunts [130].

Predictive models such as CBOW, Continuous Skip-gram, and FastText encode information by predicting words from local context [127].

CBOW is a non-task-specific pre-training objective where a feed-forward neural network processes a large corpus. The neural network predicts the centre word from the average word embedding of R context words before and after the centre word. Figure 8.1 shows the CBOW model architecture where $R = 2$. The architecture consist of two randomly initialised word embedding matrices, $\mathrm{W}_{d\times V}$ and $\mathrm{W'}_{d\times V}$. The input is sets of context words and centre word from the corpus, captured with a sliding window. During training, context words from W

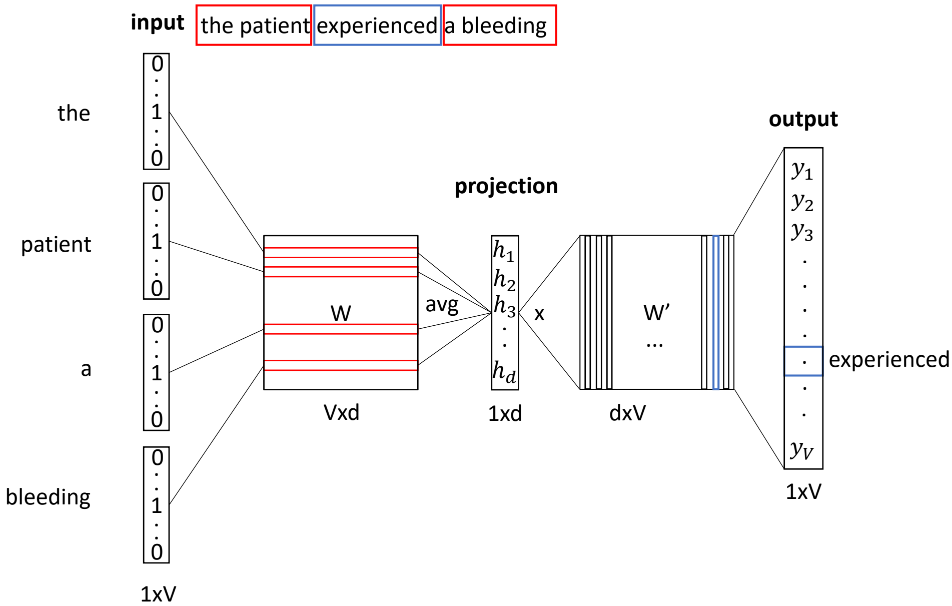

FIGURE 8.1: The Continuous Bag-of-Words model architecture. One-hot vectors are used to select the context words in the W embedding matrix, and the embeddings are averaged. Then the dot product between the projection and all word embeddings in W' is calculated. The word embedding matrices are optimized to give a high probability for the centre word. Red box: context word. Blue box: centre word. Inspired by Mikolov et al. [64] and Weng [131]. Abbreviations: V, size of vocabulary; d, dimension of word embeddings.

and centre words from W' are optimized to have high dot product similarity. This results in words with similar context having similar word embeddings, which follows the distributional hypothesis. After optimisation, language information is embedded in the word embeddings so that similar words appear close in the d-dimensional word embedding space and different words far apart [64].

The Continuous Skip-gram method applies the reversed CBOW objective and predicts multiple context words from one centre word [64]. The FastText method modifies Continuous Skip-gram to consider word morphology by representing words as a bag of character n-grams. Embeddings for each n-gram are learned, and words are represented as the sum of their n-grams [65].

Count-based models such as GloVe use global co-occurrence statistics to encode information. Instead of using a sliding window, GloVe captures global co-occurrence counts in a single sweep of the corpus and trains a model on the co-occurrence matrix [66].

8.3 Training and Benchmarking Danish Clinical Word Embeddings

This section presents our work on developing a Danish clinical word embedding benchmark that can be used to ensure that word embeddings perform optimally for the clinical domain. Furthermore, it documents the training of two Danish clinical word embedding models that can be used to create a dense clinical annotation corpus. The work is detailed in the following paper:

[50] Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Just Vinholt, Rasmus Søgaard Hansen, and Thiusius Rajeeth Savarimuthu. "Benchmark for evaluation of danish clinical word embeddings". In: *Northern European Journal of Language Technology* vol. 9, no. 1, 2023

Below, I briefly present the contents of the paper in the context of this dissertation. The paper is included at the end of this section. It is a reprint of our original paper [50], which is licensed under a CC BY 4.0 license. The CC BY 4.0 license can be found at `https://creativecommons.org/licenses/by/4.0/`.

In this paper, we created a benchmark for Danish clinical word embeddings and trained and compared clinical word embeddings using the benchmark. The benchmark consists of eight intrinsic tasks, evaluating the embeddings' inherent information on syntax and semantics, and two extrinsic tasks, evaluating their performance in downstream classification tasks. We trained clinical word embeddings using the FastText and GloVe methods and compared them to general domain FastText embeddings and general practitioner domain GloVe embeddings, respectively. We trained the clinical domain embeddings on 299,718 Danish EHRs. The general and general practitioner domain embeddings were acquired externally.

The paper contributes to this dissertation with two Danish clinical word embedding models trained using the FastText and GloVe methods and a benchmark to evaluate Danish clinical word embeddings. The benchmark can be used for finding the word embedding model that performs best on the clinical domain. The model can be used to expand the dictionary of words to match samples for inclusion in the dense annotation corpus.

-

Benchmark for Evaluation of Danish Clinical Word Embeddings

Martin S. Laursen*, University of Southern Denmark, Odense, Denmark msla@mmmi.sdu.dk
Jannik S. Pedersen*, University of Southern Denmark, Odense, Denmark jasp@mmmi.sdu.dk
Pernille Just Vinholt, Odense University Hospital, Denmark pernille.vinholt@rsyd.dk
Rasmus Søgaard Hansen, Odense University Hospital, Denmark rasmus.sogaard.hansen@rsyd.dk
Thiusius Rajeeth Savarimuthu, University of Southern Denmark, Odense, Denmark trs@mmmi.sdu.dk

Abstract In natural language processing, benchmarks are used to track progress and identify useful models. Currently, no benchmark for Danish clinical word embeddings exists. This paper describes the development of a Danish benchmark for clinical word embeddings. The clinical benchmark consists of ten datasets: eight intrinsic and two extrinsic. Moreover, we evaluate word embeddings trained on text from the clinical domain, general practitioner domain and general domain on the established benchmark. All the intrinsic tasks of the benchmark are publicly available[1].

1 Introduction

Word embeddings are real-valued vectors that are trained to represent words based on the context in which they appear. Based on the distributional hypothesis (Harris, 1954), which suggests that words with similar contexts have similar meaning, embeddings of semantically similar words are expected to appear close to each other in vector space.

Since their introduction, word embeddings have been ubiquitous in natural language processing (NLP) due to their ability to represent word meaning. Typically, word embeddings are trained on a general text corpus such as Wikipedia. Afterwards, word embeddings are used as stand-alone features or as input to neural networks to perform a wide variety of NLP tasks such as text classification, named entity recognition (NER) and machine translation.

In specialized domains, such as the clinical, word embeddings are also widely used to e.g. extract information from electronic health records (EHRs). However, the text in clinical EHRs differs significantly from the general domain. Clinical EHRs include rare words, domain specific abbreviations and a mix of languages (for example Latin, English and Danish). The text is often non-narrative and very concise, free of syntactic rules, sometimes consisting of a sequence of keywords. Moreover, it contains many spelling errors, and the semantic meaning of words can differ from that of the general domain (Leaman et al., 2015). In the clinical domain, word embeddings are, therefore, often trained on an in-domain corpus to better capture the vocabulary and the semantic meaning of words. After being trained on an in-domain corpus, they are used for e.g. clinical NER, International Classification of Diseases coding, clinical event detection, de-identification and patient similarity estimation with improved performance over general word embeddings (Zhao et al., 2018; Wang et al., 2018; Chen et al., 2019).

For evaluating word embeddings, two different methods are typically used: intrinsic and extrinsic evaluation (Wang et al., 2019c). In intrinsic evaluation, word embeddings are evaluated based on their inherent information, e.g. by exploring the syntactic or semantic relationship between words. In extrinsic evaluation, word embeddings are evaluated based on their ability to solve a downstream task, e.g. by using them as input to a neural network. While word embeddings can be evaluated using extrinsic benchmarks by holding the network architecture fixed while varying the set of word embeddings, intrinsic benchmarks provide an intermediate evaluation of the embeddings' properties before being used as input to a larger system. This supports the need for intrinsic evaluation.

Word embeddings for the general domain are publicly available in many languages (Grave et al., 2018). However, publicly available embeddings for the clinical domain are scarce (Khattak et al., 2019). This is

*Both authors contributed equally to this paper.
[1] www.github.com/jannikskytt/DaClinWordEmbeddings

most likely due to strict regulations around clinical data which contain sensitive information making them unsuitable for sharing. Therefore, researchers in clinical NLP are often forced to create their own word embeddings in order not to expose sensitive information (Abdalla et al., 2020).

Clinical intrinsic benchmark datasets do not necessarily contain sensitive information and can, in that case, be shared openly, benefitting researchers producing clinical word embeddings. For the English language, both intrinsic and extrinsic benchmarks exist, e.g. University of Minnesota Medical Residents Similarity / Relatedness Set (UMNSRS) (Pakhomov et al., 2010) for word similarity and relatedness, and BLUE (Peng et al., 2019), which includes both clinical and biomedical datasets, for extrinsic evaluation. For Danish, though, no clinical benchmark exists.

In this paper, we introduce a clinical word embedding benchmark for the Danish language. Moreover, we produce clinical word embeddings and use the benchmark to compare them to embeddings trained on the general domain and embeddings trained on the general practitioner (GP) domain.

The benchmark is specifically constructed to evaluate static word embeddings such as GloVe (Pennington et al., 2014), Continuous Bag-of-Words (Mikolov et al., 2013a), Skip-gram (Mikolov et al., 2013a) and FastText (Bojanowski et al., 2017). It is therefore not suitable for evaluation of contextual word embeddings produced by transformer models like BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) and ELECTRA (Clark et al., 2019).

Although transformer models achieve state-of-the-art results (Wang et al., 2019b; Wang et al., 2019a), static word embeddings are still useful as input to NLP pipelines. Some advantages are that they require less compute to train and at inference time, and they work better on limited data (Peng et al., 2021). This is relevant for research within specialized domains, such as clinical NLP, where researchers must often train their own word embeddings on limited data and the hardware to train and run a transformer model is not necessarily available. Static word embeddings are also relevant in time-critical tasks in clinical practice such as expanding single-word searches in the EHR using the nearest neighbors of the search term. Expanding single-word searches is especially relevant in the clinical domain where many different terms can be used about the same basic symptom or disease. Another advantage is their ease of use for medical doctors (MDs) and clinical researchers who are not machine learning scientists compared to contextual word embeddings.

The remainder of this paper first introduces the benchmark including the methods for creating each intrinsic and extrinsic dataset. It then describes the training methods of the produced Danish clinical word embeddings and those from the general and GP domains which they will be benchmarked against. Finally, the benchmark results are presented and discussed.

2 Establishing Benchmark

The benchmark consists of an intrinsic and extrinsic part. In this paper, intrinsic performance is evaluated based on the quality of the semantic and syntactic inherent information using analogy and similarity tasks. We produce datasets for three different intrinsic evaluation methods: analogy tasks, similarity and relatedness tasks, and an equality task.

Analogy tasks, introduced by Mikolov et al. (2013b), take tuples of four words (A, B, C and D) and evaluate 'what is to C as B is to A' by selecting the nearest neighbour to the calculated vector in the embedding space, excluding the words forming the analogy:

$$\overrightarrow{C} + \overrightarrow{B} - \overrightarrow{A} = \overrightarrow{D}$$

If the nearest neighbor to the calculated vector is D, the analogy task is correct. The task is evaluated on the percentage of correct predictions in the dataset.

Similar to Pennington et al. (2014), the similarity tasks take a tuple of two words and their similarity score, in our case, produced by one or more MDs. The similarity score of a word pair is compared to the cosine similarity of the pair's word embeddings. The cosine similarity is calculated as:

$$similarity(\overrightarrow{v}, \overrightarrow{u}) = \frac{\overrightarrow{v} \cdot \overrightarrow{u}}{\|\overrightarrow{v}\| \cdot \|\overrightarrow{u}\|}$$

The correlation between MD scores and cosine similarities for the dataset of word pairs is evaluated using the Spearman's rank correlation coefficient. Relatedness tasks are identical to similarity tasks except the MDs produce a relatedness score instead of a similarity score. Relatedness refers to one word calling to mind another word (e.g., needle–thread), while similarity reflects the degree of semantic feature overlap between words (e.g., whale–dolphin) (Pakhomov et al., 2010).

Equality tasks take a tuple of two terms with the exact same meaning. As the similarity score of a pair is 1 for a perfect match, the objective is maximization of the cosine similarity between terms. The task is evaluated as the mean of the cosine similarities for all pairs in the dataset.

The extrinsic part consists of two different text classification tasks in the clinical domain with the word embeddings as input. The quality of the word embeddings is evaluated based on the evaluation metric of the classification task.

An overview of all datasets can be seen in Table 1.

Task	Description	Example
Intrinsic tasks		
Clinical analogy	Evaluate "what is to C as B is to A"	$\overrightarrow{joint} + \overrightarrow{colonoscopy} - \overrightarrow{colon} = \overrightarrow{arthroscopy}$
Clinical similarity UMNSRS similarity UMNSRS relatedness	Compare the human similarity/relatedness score of a word pair to the cosine similarity of the pair's word embeddings	uterus, cervix
Clinical abbreviation equality	Compare the similarity of a word and its abbreviation	cm, centimeter
Verb, adjective, and noun inflection analogy datasets	Evaluate "what is to C as B is to A"	$\overrightarrow{adjusted} + \overrightarrow{remove} - \overrightarrow{removed} = \overrightarrow{adjust}$
Extrinsic tasks		
Bleeding classification	Classify a paragraph as either positive or negative for bleeding	15-year-old girl hospitalized with bleeding tendency and anemia symptoms
Hospital department classification	Classify a paragraph into one of six hospital departments	Clinical contact. Prepared by clinic. Conclusion and plan: As agreed and as a follow-up to the note on 10.2.99, I have contacted pt. However, pt. is hospitalized due to ...

Table 1: Overview of the datasets used in the benchmark. Examples are translated to English.

2.1 Intrinsic Datasets

The intrinsic part consists of the following semantic tasks: clinical analogy, clinical similarity, clinical abbreviation equality, and UMNSRS similarity and relatedness; and the following syntactic tasks: verb inflection analogy, adjective inflection analogy, and noun inflection analogy. The intrinsic syntactic tasks are evaluating the syntactic properties of word embeddings in general rather than specifically for clinical use cases. As good clinical word embeddings must also contain syntactic information, the syntactic tasks are constructed to specifically evaluate the inherent syntactiv information on words from the clinical domain.

The development of each intrinsic task consisted of 1) selecting the terms to use for the task and 2) creating the evaluation dataset. This is described for each task below. All intrinsic datasets are supplied in the supplementary material.

2.1.1 Clinical Analogy Dataset

Two MDs, in agreement, created 41 distinct clinical analogies such as (translated from Danish)

$$\overrightarrow{colonoscopy} - \overrightarrow{colon} = \overrightarrow{arthroscopy} - \overrightarrow{joint}$$

where the word pairs on each side of the equation have the same one-to-one relationship. For the example above with the one-to-one relationship 'is telescopic examination of', it means that colonoscopy is a telescopic examination of only the colon, and that the colon has only one telescopic examination: a colonoscopy. Some other common relationships were 'treats', 'is indicator for', 'is disease in anatomy', 'is test for', 'is examination of', 'leads to' and 'is symptom of'. We relaxed the one-to-one relationship condition in a few cases: if for example a symptom is predominant for one disease but also minorly associated with another, we accepted the word pair. We augmented each distinct analogy to form four analogies by changing the order of the words inside the word pairs and by changing the order of the word pairs. This means that, for the analogy example above, we predicted each of 'colonoscopy', 'colon', 'arthroscopy', and 'joint' from the remaining three words. We performed this augmentation because the analogy tasks are based on evaluating the nearest neighbour to the calculated vector. Since the surrounding embedding space for each of the four calculated vectors may vary in distance to neighbours, the result may vary depending on which of the four words is predicted.

The clinical analogy dataset consists of 164 analogies.

2.1.2 Clinical Similarity Dataset

For the clinical similarity dataset, we predefined the following goals for achieving a diverse set of word pairs:

1. The selected words should be of different categories, e.g. they should not all be diseases.
2. The selected words should appear with varying frequency in clinical EHRs.
3. Word pairs should be matched within and across the categories and frequencies.
4. Words should not be selected based on an existing clinical EHR database because it could introduce bias to the dataset, e.g. the frequency of

words in our clinical EHR database might differ from other databases.

To achieve this, we predefined five clinical categories: anatomy, symptom/finding, disease, treatment, and diagnostic; and three frequency categories indicating how frequently a word appears in clinical EHRs: infrequent, occasional, and frequent. Then, two MDs selected words from a reference work on internal medicine (Schaffalitzky de Muckadell et al., 2009) by turning to approximately every fifth page, randomly selecting words, and subjectively assigning them categories until all three frequency categories per five clinical categories had 36 words each. This generated a total of 108 words per clinical category and 540 words overall.

We defined 270 word pairs by pairing 36 words from each clinical category with 36 words from the same category and 36 words evenly distributed on the four other clinical categories. We opted to use more words per group for intra-category-pairs than inter-category-pairs because we expected it would decrease the overrepresentation of pairs with low similarity. The pairings were distributed evenly across frequency categories. Finally, to further decrease overrepresentation of pairs with low similarity, the MDs subjectively defined 19 extra pairs with high similarity by pairing any two words from the word pool, resulting in a total of 289 word pairs.

Ten MDs with 2 to 17 years (mean: 7.5 years) of clinical experience used between 17 and 45 minutes (mean: 30.5 minutes) to rate the 289 pairs. Nine MDs had clinical biochemistry as speciality and one had pathology. The pairs were rated for similarity on a scale from 0 to 6 with 0 being lowest similarity and 6 being highest similarity. It was emphasized that the MDs should rate for similarity and not relatedness. If a word pair was unknown to the MDs, they did not rate it. One pair was rated by eight MDs and the rest were rated by at least nine. The similarity score for each pair is the mean rating. The mean ratings span from 0 to 6 with a minimum similarity score of 0.3, a mean of 1.1, and a maximum of 5.4. The standard deviations range from 0.3 to 1.6 with a mean of 0.7.

2.1.3 Clinical Abbreviation Equality Dataset

A list of 319 clinical abbreviations and their corresponding words was collected from online sources (supplementary material). Only abbreviations of single words were collected to simplify the evaluation of word embeddings, which usually represent single words. Ambiguous abbreviations and the abbreviations deemed unlikely to appear in clinical EHRs by an MD were removed. For example, the abbreviation 'all' is ambiguous because it could both mean 'allergy' or 'acute lymphocytic leukemia'. The final dataset comprises 195 abbreviation–word pairs with the same meaning.

2.1.4 UMNSRS Similarity and Relatedness Datasets

The UMNSRS consists of 566 English term pairs rated for semantic similarity and 587 for semantic relatedness on a continuous scale from 0 to 1600. One MD translated the datasets into Danish. Pairs consisting of a term that translates into a multi-word expression were removed. As were terms that do not exist in Danish, for example a non-traded drug. In cases where a Danish counterpart drug exists, for example 'betalaktam' for 'cefoxitin', this term was used as a translation. The Danish translation of the UMNSRS consists of 528 similarity pairs and 557 relatedness pairs.

2.1.5 Verb Inflection Analogy Dataset

A list of all verbs was extracted from the Danish orthographic dictionary (Danish Language Council, 2012). One MD selected verbs from the list that were deemed would occasionally or frequently occur in a clinical EHR. Next, verbs were conjugated in the following inflections: infinitive, present/future (same form in Danish), past tense, and present/past perfect. If a verb did not exist in all four inflections or had the same form in multiple inflections, it was removed from the list as it would cause analogy tasks involving the zero-vector. The final list contained 92 words, each in four inflections.

For each verb, six types of inflection pairs were made, for example infinitive–past, by pairing each inflection with the three other inflections. Next, we randomly combined each verb with 20 other verbs, evenly distributed on types of inflection pairs except for the remainder after equal division. This produced 1,840 analogies like the following of type infinitive–past (translated from Danish):

$$\overrightarrow{remove} - \overrightarrow{removed} = \overrightarrow{adjust} - \overrightarrow{adjusted}$$

2.1.6 Adjective Inflection Analogy Dataset

The same method as described for the verb inflection analogy dataset was used to develop the adjective inflection analogy dataset. Adjectives were declined in the following inflections: common positive, neuter positive, plural positive, comparative and superlative. The final list contained 43 words, each in five inflections.

For each adjective, we made seven types of inflection pairs by pairing each of the three positive inflections with comparative and superlative and finally, the comparative with the superlative.

We combined each adjective with all other adjectives to produce 1,806 analogies.

2.1.7 Noun Inflection Analogy Dataset

We created a list from the 180 frequent words from the combined five clinical categories of the clinical similarity dataset. We removed words which were not nouns and declined the remaining in the following inflections: indefinite singular, definite singular, indefinite plural and definite plural. If a noun did not exist in all four inflections or had the same form in multiple inflections, it was removed from the list. The final list contained 138 words, each in four inflections. For each noun, we made six types of inflection pairs by pairing each inflection with the three other inflections. Next, we randomly combined each noun with 13 other nouns, evenly distributed on types of inflection pairs except for the remainder after equal division, to produce 1,794 analogies.

2.2 Extrinsic Datasets

The extrinsic part consists of a hospital department classification task and a bleeding classification task. All datasets were obtained according to each dataset's respective data usage policy. The datasets are described below.

2.2.1 Bleeding Classification

For the bleeding classification dataset, we used that of Pedersen et al. (2021). It consists of 9,430 training sentences, 1,178 validation sentences, and 1,178 test sentences which are evenly distributed on the two classes: 'indicates bleeding' and 'does not indicate bleeding'. The latter class consists of 50% sentences that were deemed by the MDs to be at high risk of being misinterpreted by the deep learning model. The other 50% were random negative sentences. The classification objective is to predict if a sentence indicates bleeding.

The data came from 300 EHRs corresponding to 88,477 notes from the EHR system of the Region of Southern Denmark between 2015 and 2020. The sentences were annotated by splitting the annotation of EHRs between twelve MDs.

2.2.2 Hospital Department Classification

The hospital department classification dataset was constructed without the need of human annotators by using the department associated with each note as a label. This approach is an advantage since the task of annotating clinical records is time consuming and expensive.

The hospital department classification dataset consists of 42,000 clinical EHR notes evenly distributed on the following six Odense University Hospital departments: Cardiology; Cardiac, Thoracic and Vascular Surgery; Orthopaedic Surgery; Rheumatology; Surgery; and Medical Gastrointestinal Diseases. Danish clinical EHR notes have a tree structure consisting of many generic node headlines. MDs only fill out the end-nodes manually. To avoid node headlines or text passages specific to one department making the classification a simple task, each note was preprocessed by only keeping the lowercased end-node texts. Furthermore, end-nodes which were duplicates based only on their words, disregarding all but letters, were removed across the whole dataset. The notes are between 51 and 220 tokens. The dataset contains 7,000 notes from each department in a class-balanced train:validation:test ratio of 5:1:1. The classification objective is to predict the hospital department.

3 Word Embedding Evaluation

This section describes an evaluation of word embedding models, trained on data from different domains, using the established benchmark. We make a clinical–general domain comparison using a FastText (Bojanowski et al., 2017) model as it has the best performance on Danish text according to benchmark results (Brogaard Pauli et al., 2021). We make a clinical–GP domain comparison using a GloVe (Pennington et al., 2014) model as it is the only available type of embeddings trained on Danish GP data. We describe how the benchmark can be used to show strengths and weaknesses of different word embeddings.

We trained two sets of clinical word embeddings using the FastText and GloVe methods. The embeddings were trained on 299,718 Danish EHRs from Odense University Hospital. The text was preprocessed by lowercasing and removing headlines, subheadings, phone numbers, social security numbers, emails, URLs, dates and time stamps. Samples were defined as text from the same subheading. After removal of duplicates and samples with less than 3 words, the corpus consisted of 1.4 billion tokens.

For the clinical–general domain comparison, the clinical FastText embeddings were trained with the default settings from the FastText API (www.fasttext.cc) except from a vector size of 300, 10 negative samples and 10 epochs. The hyperparameters were chosen to be able to compare the produced embeddings with the FastText word embeddings from Grave et al. (2018) pretrained on a general domain, specifically Wikipedia and Common Crawl. The FastText models can generate out-of-vocabulary (OOV) words from subwords which e.g. makes it capable of representing unknown spelling errors. For clarity, only the results without OOV generation are reported here while the results with OOV generation are found in Appendix A.

For the clinical–GP domain comparison, the clinical GloVe embeddings are 100-dimensional embeddings trained with the default settings from the code and paper by Pennington et al. (2014) except for a min-count of 3. The hyperparameters were chosen to be able to compare with the GloVe word embeddings from Rasmussen et al. (2019) trained on 323,122 GP EHRs.

The word embedding models are benchmarked on the established intrinsic and extrinsic datasets. For each intrinsic task, we show the performance of the embeddings on the part of the evaluation dataset which is in-vocabulary (IV), ignoring the word pairs or analogies containing OOV words. We also produce the IV rate as the proportion of word pairs or analogies which are in the vocabulary of the embeddings. Additionally, Appendix B contains the IV intersection results which show the performance of the embeddings on the intersection of all embeddings' IV dataset for that task.

For the extrinsic tasks, the word embeddings are used as input to a recurrent neural network which is initialized and trained three times with the same set of standard hyperparameters. No hyperparameter tuning is performed. A bidirectional gated recurrent unit (Cho et al., 2014) with 128 units followed by a dropout layer with probability 0.3 is trained with the Adam optimizer with a learning rate of 5e-4 for a maximum of 100 epochs using early stopping. The best model, based on the validation loss, is evaluated on the test set. The test set accuracy is reported as the evaluation result.

3.1 Intrinsic Results

We present the intrinsic semantic and syntactic benchmark results.

3.1.1 Semantic Results

Table 2 shows the intrinsic semantic results. The clinical FastText embeddings achieve better performance than the general FastText embeddings on the abbreviation equality task, clinical similarity task, UMNSRS similarity task and UMNSRS relatedness task. The clinical analogy task shows different results with the general FastText embeddings performing better with an IV accuracy of 0.14 while the clinical FastText embeddings have an IV accuracy 0.05. The clinical GloVe embeddings perform better than the GP GloVe embeddings on all intrinsic semantic tasks.

The word embeddings trained on the clinical domain show the highest IV rates, followed by the GP domain and then the general domain. The two clinical models have an IV rate equal to or higher than 0.83 for all semantic tasks. The GP GloVe embeddings have IV rates between 0.57 and 0.75 while the general FastText embeddings have IV rates between 0.54 and 0.61.

Appendix C presents the correct clinical analogy predictions for all word embedding models. Moreover, Appendix D shows the results on the clinical analogy task where a prediction is considered correct if the correct term is in the top 1, 5 and 10 nearest neighbours to the calculated vector.

3.1.2 Syntactic Results

Table 3 shows the intrinsic syntactic results. The results show that the general FastText embeddings achieve better performance than the clinical FastText embeddings on all syntactic tasks with an IV accuracy of 0.69 on verbs, 0.60 on nouns and 0.41 on adjectives. The clinical FastText embeddings perform at IV accuracies of 0.28, 0.19 and 0.16, respectively. The clinical GloVe embeddings perform better than the GP GloVe embeddings on the verb and noun inflection tasks with IV accuracies of 0.21 and 0.04, and 0.09 and 0.01, respectively. The GP GloVe embeddings perform best on the adjective inflection task with an IV accuracy of 0.04 contra 0.03 for the clinical GloVe embeddings.

The clinical domain embeddings have the highest IV rates for the verb and noun inflection tasks at 0.99 and 0.39, respectively. The general FastText embeddings have the highest IV rate for the adjective inflection task at 0.65, followed by the clinical GloVe embeddings at 0.47.

3.2 Extrinsic Results

Table 4 shows the extrinsic results. For both the FastText and GloVe models, the clinical domain embeddings achieve higher performances than their respective general domain and GP domain counterparts.

4 Discussion

In this paper, we have presented the first benchmark for evaluating Danish clinical word embeddings. Although the clinical word embeddings cannot be shared due to privacy concerns, having a publicly available benchmark will allow researchers to compare and evaluate locally available clinical word embeddings. Below, we discuss the capability of the benchmark to compare word embedding performance in the clinical domain.

As the intrinsic benchmark tasks consist of words which are typically, and in some cases, exclusively, used in the clinical domain, we expected higher IV rates from clinical domain embeddings. In concurrence, the results show that the clinical word embeddings, in general, have higher IV rates than those trained on the GP and general domain. An exception is that the general FastText embeddings have the highest IV rate for the adjective inflection analogy task. One explanation could be that clinical written language does not use as many inflections of adjectives as the general. Interestingly, when comparing the GP and general word em-

	FastText (300d)	
	Clinical	**General**
Clinical analogy, accuracy (IV)	0.05 (0.88)	<u>0.14</u> (0.54)
Abbreviation equality, similarity (IV)	<u>0.53</u> (0.84)	0.27 (0.58)
Clinical similarity, ρ (IV)	<u>0.64</u> (0.93)	0.43 (0.61)
UMNSRS similarity, ρ (IV)	<u>0.60</u> (0.88)	0.30 (0.59)
UMNSRS relatedness, ρ (IV)	<u>0.54</u> (0.83)	0.32 (0.56)
	GloVe (100d)	
	Clinical	**GP**
Clinical analogy, accuracy (IV)	<u>0.08</u> (0.88)	0.06 (0.61)
Abbreviation equality, similarity (IV)	<u>0.49</u> (0.85)	0.24 (0.57)
Clinical similarity, ρ (IV)	<u>0.56</u> (0.96)	0.34 (0.75)
UMNSRS similarity, ρ (IV)	<u>0.41</u> (0.89)	0.18 (0.74)
UMNSRS relatedness, ρ (IV)	<u>0.41</u> (0.84)	0.21 (0.70)

Table 2: Semantic benchmark results on the in-vocabulary (IV) dataset for each task by model type (FastText, GloVe) and domain (clinical, general, general practitioner (GP)). The accuracy metric is the accuracy on the dataset. The similarity metric is the average cosine similarity on the dataset. The ρ metric is the Spearman's rank correlation coefficient on the dataset. IV rates are reported in parenthesis. We underline the best results per task by model type.

	FastText (300d)	
	Clinical	**General**
Verb inflection analogy, accuracy (IV)	0.28 (0.99)	<u>0.69</u> (0.92)
Noun inflection analogy, accuracy (IV)	0.19 (0.36)	<u>0.60</u> (0.13)
Adjective inflection analogy, accuracy (IV)	0.16 (0.36)	<u>0.41</u> (0.65)
	GloVe (100d)	
	Clinical	**GP**
Verb inflection analogy, accuracy (IV)	<u>0.21</u> (0.99)	0.09 (0.83)
Noun inflection analogy, accuracy (IV)	<u>0.04</u> (0.39)	0.01 (0.18)
Adjective inflection analogy, accuracy (IV)	0.03 (0.47)	<u>0.04</u> (0.25)

Table 3: Syntactic benchmark results on the in-vocabulary (IV) dataset for each task by model type (FastText, GloVe) and domain (clinical, general, general practitioner (GP)). The accuracy metric is the accuracy on the dataset. IV rates are reported in parenthesis. We underline the best results per task by model type.

beddings on the semantic tasks, the GP embeddings, in four out of five tasks, have higher IV rates but lower accuracy. This result shows that the GP embeddings have seen more clinical domain words than the general embeddings during training, but the general embeddings capture higher quality information for the words that it has seen. This could be due to the size and quality of the dataset, differences between model types or the dimensionality of the embeddings. Future work should investigate these claims further.

The benchmark shows that the clinical embeddings surpass the general and GP embeddings in all semantic tasks except for the clinical analogy task where the general FastText embeddings performed better than the clinical FastText embeddings. This discrepancy may be caused by the clinical analogy dataset only containing 164 analogies of which only 54% are IV for the general FastText model.

The general embeddings surpass the clinical embeddings on the syntactic tasks which shows that it has captured higher quality syntactic information for the words that it has seen during training. This is most likely due to Wikipedia and Common Crawl, which it was trained on, containing a higher quality of syntactic information than clinical EHRs.

The fact that the general embeddings achieve the highest IV rate on the adjective inflection task suggests that the task consists of more inflections specific to the general domain than our clinical dataset. On the contrary, clinical domain embeddings achieve the highest IV rates on the verb and noun inflection tasks which suggests that these syntactic tasks do contain inflections specific to the clinical domain.

Similar to earlier work (Zhao et al., 2018; Wang et al., 2018; Chen et al., 2019), we found that the clinical word embeddings perform better than the GP and general domain embeddings on extrinsic tasks. It is notable that for the extrinsic tasks, the GP GloVe embeddings are closer to the performance of the clinical GloVe embeddings than the general FastText embeddings are to

	FastText (300d)	
	Clinical	**General**
Bleeding classification, accuracy	<u>0.93</u>	0.84
Department classification, accuracy	<u>0.83</u>	0.65
	GloVe (100d)	
	Clinical	**GP**
Bleeding classification, accuracy	<u>0.90</u>	0.87
Department classification, accuracy	<u>0.76</u>	0.66

Table 4: Extrinsic benchmark results by model type (FastText, GloVe) and domain (clinical, general, general practitioner (GP)). We report accuracies on the class-balanced bleeding and department classification tasks using the word embeddings as input. We underline the best results per task by model type.

that of the clinical FastText embeddings. This could be explained by the fact that there is some similarity between the GP and clinical domains, both being subdomains of the healthcare domain.

Considering that the general embeddings perform well on syntactic tasks and clinical embeddings perform well on semantic and extrinsic tasks, future work should explore training word embeddings from the general FastText checkpoint on clinical data. This might provide word embeddings that better capture both clinical and general syntactic and semantic properties.

4.1 Limitations

This study compared GloVe and FastText word embeddings. While FastText performed best on some benchmarks, other word embedding methods might perform better. We leave these investigations to future work.

Future use of the presented resources relies on the assumption that the words in the intrinsic datasets also appear in the user's vocabulary. In section 2.1.2 we described how we tried to mitigate this shortcoming.

The clinical similarity dataset would benefit from including more pairs with high similarity and decreasing the mean standard deviation, e.g. by including more raters from different specialities. To alleviate MD rating disagreement, we have included in the supplementary material the clinical similarity ratings for each MD with information about the standard deviation of each word pair, which can be used to set a threshold of maximum allowed disagreement. Appendix E shows the results on the clinical similarity dataset consisting of pairs with standard deviations at or below 1.

The extrinsic department classification task might as well classify the writing styles of specific MDs in a department, thus not necessarily generalizing to other MDs. This can be remedied by having unique authors in the test split.

It is a limitation to the extrinsic results that no hyperparameter tuning was performed. Results from a model trained with a standard set of hyperparameters can rank the word embeddings but the results are not indicative of the best performance of each embedding.

We have shown a discrepancy between the clinical analogy task and all other semantic tasks. We believe it would be beneficial to include more analogies in the clinical analogy dataset as the result is based on few IV analogies.

The syntactic results suggest that the adjective inflection task consists of more inflections specific to the general domain than the clinical domain. Many of the inflections do exist in the clinical domain but it is a limitation for the evaluation of clinical word embeddings that not enough inflections are specific for the clinical domain.

The tasks were designed to evaluate static word embeddings using only single-word expressions which limits the use of the benchmark for contextual word embeddings such as tranformer models and word embeddings trained on n-grams.

It is a limitation to our benchmark that it only provides two extrinsic tasks, and in general, that there are no Danish clinical extrinsic datasets publicly available. Due to privacy concerns, we cannot publish the extrinsic datasets, but we provide a method for creating an extrinsic test that leverages already existing labels in the form of the department of the clinical note. This method does not need any labeling but still requires access to EHRs. We encourage interested researchers to contact us for the possibility of sharing the extrinsic datasets.

Future work should focus on developing more diverse extrinsic tasks such as named entity recognition, relation extraction and question answering.

5 Conclusion

In this paper, we presented a benchmark for Danish clinical word embeddings. The benchmark consists of two extrinsic tasks, five intrinsic semantic tasks and three intrinsic syntactic tasks. We developed clinical word embeddings and compared them with word embeddings trained on a general and general practitioner domain. The benchmark showed that the word embeddings trained on clinical data performed better on the extrinsic and semantic tasks, except for the clinical analogy task. On the syntactic tasks, the FastText word embeddings trained on a general domain performed better than those trained on a clinical domain.

Acknowledgements

The authors thank Anne Bryde Alnor, Charlotte Gils, Eline Sandvig Andersen, Ida Stangerup, Jesper Dupont Ewald, Jesper Farup Revsholm, Katrine Sølling Borlund Madsen and Kristina Bjerg Appel for the rating work for the clinical similarity task.

References

Abdalla, Mohamed, Moustafa Abdalla, Graeme Hirst, and Frank Rudzicz. 2020. Exploring the privacy-preserving properties of word embeddings: Algorithmic validation study. *J Med Internet Res*, 22(7):e18055.

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Brogaard Pauli, Amalie, Maria Barrett, Ophélie Lacroix, and Rasmus Hvingelby. 2021. DaNLP: An open-source toolkit for danish natural language processing. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa 2021)*.

Chen, Qingyu, Yifan Peng, and Zhiyong Lu. 2019. Biosentvec: creating sentence embeddings for biomedical texts. In *2019 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 1–5.

Cho, Kyunghyun, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Clark, Kevin, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Danish Language Council. 2012. *Retskrivningsordbogen*, 4 edition. Danish Language Council. Including 8 digital issues (2017).

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Grave, Edouard, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Harris, Zellig S. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Khattak, Faiza Khan, Serena Jeblee, Chloé Pou-Prom, Mohamed Abdalla, Christopher Meaney, and Frank Rudzicz. 2019. A survey of word embeddings for clinical text. *Journal of Biomedical Informatics*, 100:100057. Articles initially published in Journal of Biomedical Informatics: X 1-4, 2019.

Leaman, Robert, Ritu Khare, and Zhiyong Lu. 2015. Challenges in clinical natural language processing for automated disorder normalization. *Journal of Biomedical Informatics*, 57:28–37.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Schaffalitzky de Muckadell, Ove B., Stig Haunsø, and Hendrik Vilstrup. 2009. *Medicinsk kompendium*, 17 edition. Nyt Nordisk Forlag.

Pakhomov, Serguei, Bridget McInnes, Terrence Adam, Ying Liu, Ted Pedersen, and Melton Genevieve B. 2010. Semantic similarity and relatedness between clinical terms: An experimental study. *AMIA annual symposium proceedings*, 2010:572–576.

Pedersen, Jannik S., Martin S. Laursen, Thiusius Rajeeth Savarimuthu, Rasmus Søgaard Hansen, Anne Bryde Alnor, Kristian Voss Bjerre, Ina Mathilde Kjær, Charlotte Gils, Anne-Sofie Faarvang Thorsen, Eline Sandvig Andersen, Cathrine Brødsgaard Nielsen, Lou-Ann Christensen Andersen, Søren Andreas Just, and Pernille Just Vinholt. 2021. Deep learning detects and visualizes bleeding events in electronic health records. *Research and Practice in Thrombosis and Haemostasis*, 5(4):e12505.

Peng, X, Y Zheng, C Lin, and A Siddharthan. 2021. Summarising historical text in modern languages. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3123–3142. Association for Computational Linguistics (ACL).

Peng, Yifan, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Pennington, Jeffrey, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Rasmussen, Mathias, Nichlas Berggrein, and Leon Derzcynski. 2019. Named entity recognition and disambiguation in danish electronic health records. Master's thesis, IT University of Copenhagen.

Wang, Alex, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019a. Superglue: a stickier benchmark for general-purpose language understanding systems. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 3266–3280.

Wang, Alex, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019b. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019*.

Wang, Bin, Angela Wang, Fenxiao Chen, Yuncheng Wang, and C.-C. Jay Kuo. 2019c. Evaluating word embedding models: methods and experimental results. *APSIPA Transactions on Signal and Information Processing*, 8:e19.

Wang, Yanshan, Sijia Liu, Naveed Afzal, Majid Rastegar-Mojarad, Liwei Wang, Feichen Shen, Paul Kingsbury, and Hongfang Liu. 2018. A comparison of word embeddings for the biomedical natural language processing. *Journal of Biomedical Informatics*, 87:12–20.

Zhao, Mengnan, Aaron J. Masino, and Christopher C. Yang. 2018. A framework for developing and evaluating word embeddings of drug-named entity. In *Proceedings of the BioNLP 2018 workshop*, pages 156–160, Melbourne, Australia. Association for Computational Linguistics.

A Benchmark Results Including Models With OOV Generation

Table 5 shows the semantic benchmark results for all models, including the FastText models with OOV generation.

Table 6 shows the syntactic benchmark results for all models, including the FastText models with OOV generation.

Table 7 shows the extrinsic benchmark results for all models, including the FastText models with OOV generation.

B In-Vocabulary Intersection Results

We report the IV intersection results which show the performance of the embeddings on the intersection of all embeddings' IV dataset for that task. We also report relative coverage (RC) for each model as the proportion that the IV words of a model constitute out of the union of all models' IV words.

Table 8 shows the semantic benchmark results on the intersection of IV datasets of all embeddings.

Table 9 shows the syntactic benchmark results on the intersection of IV datasets of all embeddings.

C Correctly Predicted Semantic Analogies

Table 10 shows the correctly predicted semantic analogies for all models, including the FastText models with OOV generation.

D Clinical Analogy Task Top N Accuracies

Table 11 shows the results on the clinical analogy task where a prediction is considered correct if the correct term is in the top 1, 5 and 10 nearest neighbours to the calculated vector.

E Results on Clinical Similarity Dataset With Standard Deviation at or Below 1

Table 12 shows the results on the clinical similarity dataset with standard deviation at or below 1.

F Supplementary Material

All datasets are txt files with tab-separated values. Each row has one word pair or analogy. Some datasets are divided into parts. A headline of a part is in all caps and introduced with ': '.

The following datasets are attached:

- Clinical analogy dataset (txt)
- Abbreviation equality dataset (txt)
- Clinical similarity dataset (txt)
- Clinical similarity SD1 dataset (txt)
- UMNSRS similarity dataset (txt)
- UMNSRS relatedness dataset (txt)
- Verb inflection analogy dataset (txt)
- Noun inflection analogy dataset (txt)
- Adjective inflection analogy dataset (txt)

The clinical similarity ratings and their standard deviations are found in file:

- Clinical similarity ratings (xlsx)

The online sources of clinical abbreviations are found in file:

- Abbreviation sources (xlsx)

	FastText (300d)			
	Clinical		General	
	No gen.	OOV gen.	No gen.	OOV gen.
Clinical analogy, acc (IV)	0.05 (0.88)	0.04 (1.0)	<u>0.14</u> (0.54)	0.07 (1.0)
Abbreviation equality, sim (IV)	<u>0.53</u> (0.84)	0.52 (1.0)	0.27 (0.58)	0.30 (1.0)
Clinical similarity, ρ (IV)	<u>0.64</u> (0.93)	0.62 (1.0)	0.43 (0.61)	0.32 (1.0)
UMNSRS similarity, ρ (IV)	<u>0.60</u> (0.88)	0.58 (1.0)	0.30 (0.59)	0.25 (1.0)
UMNSRS relatedness, ρ (IV)	<u>0.54</u> (0.83)	<u>0.54</u> (1.0)	0.32 (0.56)	0.27 (1.0)
	GloVe (100d)			
	Clinical (No gen.)		GP (No gen.)	
Clinical analogy, acc (IV)	<u>0.08</u> (0.88)		0.06 (0.61)	
Abbreviation equality, sim (IV)	<u>0.49</u> (0.85)		0.24 (0.57)	
Clinical similarity, ρ (IV)	<u>0.56</u> (0.96)		0.34 (0.75)	
UMNSRS similarity, ρ (IV)	<u>0.41</u> (0.89)		0.18 (0.74)	
UMNSRS relatedness, ρ (IV)	<u>0.41</u> (0.84)		0.21 (0.70)	

Table 5: Semantic benchmark results by model type (FastText, GloVe), domain (clinical, general, general practitioner (GP)), and out-of-vocabulary (OOV) generation (no gen., OOV gen.). The acc metric is the accuracy on the in-vocabulary (IV) dataset. The sim metric is the average cosine similarity on the IV dataset. The ρ metric is the Spearman's rank correlation coefficient on the IV dataset. IV rates are reported in parenthesis. We underline the best results per task by model type.

	FastText (300d)			
	Clinical		General	
	No gen.	OOV gen.	No gen.	OOV gen.
Verb inflection, acc (IV)	0.28 (0.99)	0.28 (1.0)	<u>0.69</u> (0.92)	0.66 (1.0)
Noun inflection, acc (IV)	0.19 (0.36)	0.11 (1.0)	<u>0.60</u> (0.13)	0.20 (1.0)
Adjective inflection, acc (IV)	0.16 (0.36)	0.07 (1.0)	<u>0.41</u> (0.65)	0.29 (1.0)
	GloVe (100d)			
	Clinical (No gen.)		GP (No gen.)	
Verb inflection, acc (IV)	<u>0.21</u> (0.99)		0.09 (0.83)	
Noun inflection, acc (IV)	<u>0.04</u> (0.39)		0.01 (0.18)	
Adjective inflection, acc (IV)	0.03 (0.47)		<u>0.04</u> (0.25)	

Table 6: Syntactic benchmark results by model type (FastText, GloVe), domain (clinical, general, general practitioner (GP)), and out-of-vocabulary (OOV) generation (no gen., OOV gen.). The acc metric is the accuracy on the in-vocabulary (IV) dataset. IV rates are reported in parenthesis. We underline the best results per task by model type.

	FastText (300d)			
	Clinical		General	
	No gen.	OOV gen.	No gen.	OOV gen.
Bleeding classification, acc	<u>0.93</u>	0.92	0.84	0.84
Department classification, acc	<u>0.83</u>	<u>0.83</u>	0.65	0.64
	GloVe (100d)			
	Clinical (No gen.)		GP (No gen.)	
Bleeding classification, acc	<u>0.90</u>		0.87	
Department classification, acc	<u>0.76</u>		0.66	

Table 7: Extrinsic benchmark results by model type (FastText, GloVe), domain (clinical, general, general practitioner (GP)), and out-of-vocabulary (OOV) generation (no gen., OOV gen.). We report accuracies on the bleeding and department classification task using the word embeddings as input. We underline the best results per task by model type.

	FastText (300d)	
	Clinical	**General**
Clinical analogy, accuracy (RC)	0.06 (1.0)	0.14 (0.61)
Abbreviation equality, similarity (RC)	0.55 (0.97)	0.27 (0.67)
Clinical similarity, ρ (RC)	0.67 (0.98)	0.44 (0.63)
UMNSRS similarity, ρ (RC)	0.57 (0.98)	0.28 (0.66)
UMNSRS relatedness, ρ (RC)	0.52 (0.97)	0.29 (0.66)
	GloVe (100d)	
	Clinical	**GP**
Clinical analogy, accuracy (RC)	0.13 (1.0)	0.08 (0.69)
Abbreviation equality, similarity (RC)	0.60 (0.98)	0.25 (0.66)
Clinical similarity, ρ (RC)	0.60 (1.0)	0.35 (0.79)
UMNSRS similarity, ρ (RC)	0.40 (0.99)	0.15 (0.82)
UMNSRS relatedness, ρ (RC)	0.40 (0.98)	0.23 (0.82)

Table 8: Semantic benchmark results on the intersection of IV datasets for each task by model type (FastText, GloVe) and domain (clinical, general, general practitioner (GP)). The accuracy metric is the accuracy on the dataset. The similarity metric is the average cosine similarity on the dataset. The ρ metric is the Spearman's rank correlation coefficient on the dataset. Relative coverage (RC) is reported in parenthesis. We underline the best results per task by model type.

	FastText (300d)	
	Clinical	**General**
Verb inflection analogy, accuracy (RC)	0.29 (0.99)	0.71 (0.92)
Noun inflection analogy, accuracy (RC)	0.17 (0.92)	0.63 (0.54)
Adjective inflection analogy, accuracy (RC)	0.18 (0.55)	0.54 (0.98)
	GloVe (100d)	
	Clinical	**GP**
Verb inflection analogy, accuracy (RC)	0.23 (0.99)	0.09 (0.83)
Noun inflection analogy, accuracy (RC)	0.08 (0.98)	0.03 (0.64)
Adjective inflection analogy, accuracy (RC)	0.05 (0.71)	0.04 (0.38)

Table 9: Syntactic benchmark results on the intersection of IV datasets for each task by model type (FastText, GloVe) and domain (clinical, general, general practitioner (GP)). The accuracy metric is the accuracy on the dataset. Relative coverage (RC) is reported in parenthesis. We underline the best results per task by model type.

	FastText (300d)				GloVe (100d)	
	Clinical		General		Clinical	GP
	No gen.	OOV gen.	No gen.	OOV gen.	No gen.	No gen.
hoftealloplastik + knæ - knæalloplastik = hofte (hip replacement + knee - knee replacement = hip)	✓	✓	✓	✓	✓	✓
hofte + knæalloplastik - knæ = hoftealloplastik (hip + knee replacement - knee = hip replacement)	✓	✓	✓	✓	✓	✓
knæalloplastik + hofte - hoftealloplastik = knæ (knee replacement + hip - hip replacement = knee)	✓	✓	✓	✓	✓	✓
knæ + hoftealloplastik - hofte = knæalloplastik (knee + hip replacement - hip = knee replacement)	✓	✓			✓	✓
ovarier + mand - testikler = kvinde (ovaries + man - testicles = woman)	✓	✓	✓	✓	✓	✓
testikler + kvinde - ovarier = mand (testicles + woman - ovaries = man)			✓	✓		
trombocytpool+anæmi - sag-m = trombocytopeni (thrombocyte pool + anemia - sag-m = thrombocytopenia)	✓	✓				
sag-m + trombocytopeni - trombocytpool = anæmi (sag-m + thrombocytopenia - thrombocyte pool = anemia)	✓	✓			✓	
høretab + øjne - synstab = ører (hearing loss + eyes - visual obscuration = ears)			✓	✓	✓	
synstab + ører - høretab = øjne (visual obscuration + ears - hearing loss = eyes)			✓	✓		
mad + tørst - væske = sult (food + thirst - liquid = hunger)			✓	✓	✓	
milt + gastrektomi - mavesæk = splenektomi (spleen + Gastrectomy - stomach = splenectomy)			✓	✓	✓	
aids + borrelia - neuroborreliose = hiv (aids + borreliosis - neuroborreliosis = hiv)			✓	✓		
levothyroxin + hyperthyroidisme - thiamazol = hypothyroidisme (levothyroxine + lyperthyroidism - thiamazole = hypothyroidism)			✓	✓		
respirator + nyresvigt - dialyse = respirationssvigt (respirator + renal failure - dialysis = respiratory failure)			✓	✓		
virus + dyrkning - bakterie = pcr (virus + cultivation - bacteria = pcr)					✓	
tarm + hæmoptyse - lunger = melæna (intestine + hemoptysis - lung = melena)					✓	
Kreatinin + knoglemarvsfunktion - differentialtælling = nyrefunktion (creatinine + bone marrow function - differential count = renal function)					✓	
nyrefunktion + differentialtælling - knoglemarvsfunktion = kreatinin (renal function + differential count - bone marrow function = creatinine)						✓

Table 10: Overview of the correctly predicted semantic analogies by model type (FastText, GloVe), domain (clinical, general, general practitioner (GP)), and out-of-vocabulary (OOV) generation (no gen., OOV gen.). Each analogy is presented in Danish, and the English translation is parenthesis. A mark signifies a correctly predicted analogy.

	FastText (300d)			
	Clinical		General	
	No gen.	OOV gen.	No gen.	OOV gen.
Top 1, acc	0.05	0.04	<u>0.14</u>	0.07
Top 5, acc	0.10	0.09	<u>0.27</u>	0.16
Top 10, acc	0.13	0.11	<u>0.41</u>	0.28
IV rate	0.88	1.0	0.54	1.0
	GloVe (100d)			
	Clinical (No gen.)		**GP (No gen.)**	
Top 1, acc	<u>0.08</u>		0.06	
Top 5, acc	<u>0.15</u>		0.08	
Top 10, acc	<u>0.21</u>		0.13	
IV rate	0.88		0.61	

Table 11: Top n accuracies and IV rate on the semantic clinical analogy dataset by model type (FastText, GloVe), domain (clinical, general, general practitioner (GP)), and out-of-vocabulary (OOV) generation (no gen., OOV gen.). A prediction is considered correct if the correct term is in the top n nearest neighbours to the calculated vector. The IV rate is the proportion of word pairs or analogies which are in-vocabulary. We underline the best results by model type.

	FastText (300d)			
	Clinical		General	
	No gen.	OOV gen.	No gen.	OOV gen.
Clinical similarity SD1, ρ	<u>0.60</u>	0.57	0.44	0.35
IV rate	0.94	1.0	0.61	1.0
	GloVe (100d)			
	Clinical (No gen.)		**GP (No gen.)**	
Clinical similarity SD1, ρ	<u>0.54</u>		0.40	
IV rate	0.96		0.75	

Table 12: Results on the clinical similarity dataset with standard deviation at or below 1 by model type (FastText, GloVe), domain (clinical, general, general practitioner (GP)), and out-of-vocabulary (OOV) generation (no gen., OOV gen.). The ρ metric is the Spearman's rank correlation coefficient on the IV dataset. The IV rate is the proportion of word pairs or analogies which are in-vocabulary. The dataset contains 255 word pairs. We underline the best result by model type.

8.4 Concluding Remarks

In this chapter, I described how a dense annotation corpus can provide a more equal proportion of samples from each class and thereby increase the efficiency of the annotation process. I also presented the work of our paper that introduced a Danish clinical word embedding benchmark. The benchmark can be used to find the word embedding model that performs best on the clinical domain. The work produced two Danish clinical word embedding models that was trained with the FastText and GloVe methods. The models can be used to expand the dictionary of words to match samples for inclusion in the dense annotation corpus.

Negative Samples from a Dense Corpus

Apart from ensuring a more efficient annotation process, using a dense corpus for annotation can also change the distribution of data that the model is trained on. For binary classification models like that for bleeding or VTE events, the negative samples for the training set are normally drawn randomly from the negative sentences in the corpus. By first filtering all sentences of the corpus through the rule-based filter to create a dense corpus, the distribution of negative samples changes. Samples that were accepted by the rule-based filter but manually annotated as negative contain at least one word that is associated with a positive sample by the rule-based filter. This makes the deep learning model train on a more specific task. Instead of differentiating positive samples from random negative samples, it is now trained to differentiate positive samples from negative samples that were classified by a simple rule-based system as positive. It is essentially a pipeline of two classification models where a rule-based classifier, with sensitivity prioritised over specificity, filters easy negative samples, and the deep learning model takes the positive predictions and performs a more specialised classification. This may improve the performance of the system. The principle is mirrored in the healthcare system where patients are screened and sent to the appropriate department for further testing. If all patients underwent all possible testing available at the hospital, there is a greater risk of false positive results.

Future Work

Future work includes validating and applying the method in practice. The method could, e.g. be applied to a further annotation process for the models presented in chapters 4, 5, and 6.

CHAPTER 9

Conclusion

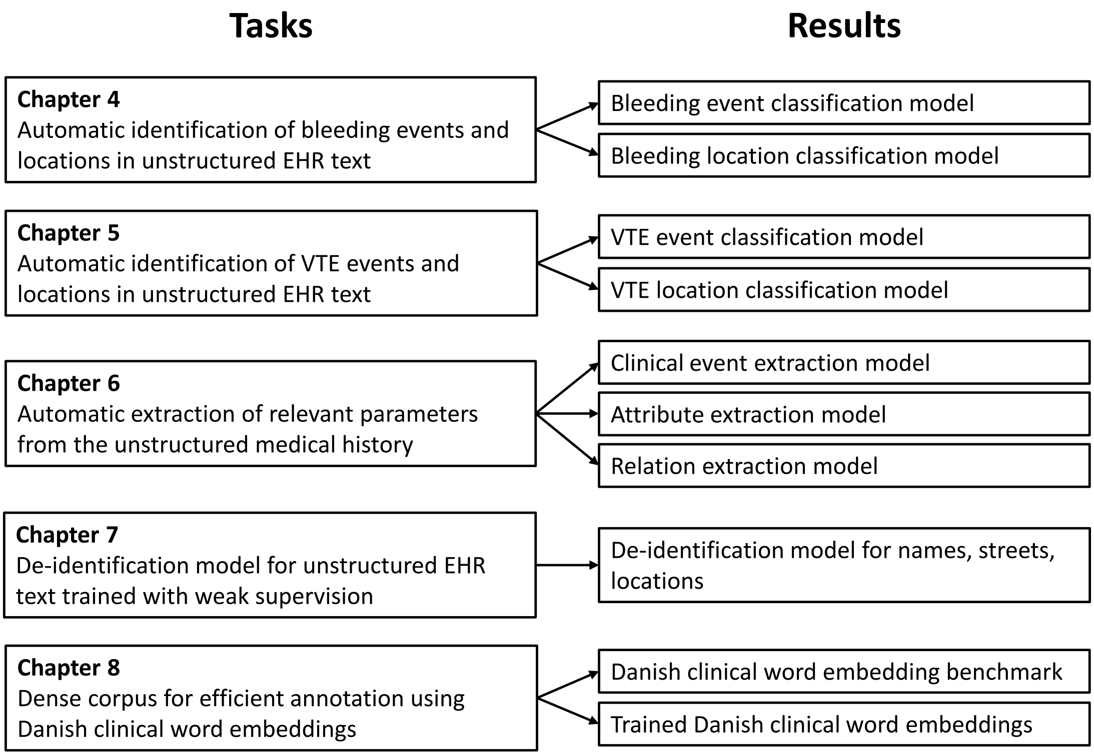

FIGURE 9.1: A repeated overview of the tasks from figure 1.2, and the results consisting of the models and tools that were developed in this dissertation. Abbreviations: EHR, electronic health record; VTE, venous thromboembolism.

Figure 9.1 contains an overview of the results including the models and tools developed in each chapter. Below, I provide a conclusion for each chapter with a focus on answering the research questions. Finally, I suggest the combined future work for this dissertation.

The overall purpose of this PhD dissertation is to enable Danish clinical research on bleeding and VTE by automatically identifying sentences that indicate these conditions; secondly, to provide a structured overview of the medical history by extracting it from the unstructured EHR text; and thirdly, to develop models to remove PII and make annotation more efficient to facilitate clinical research and model development.

Chapter 4 addressed research question 1. I presented a deep learning pipeline to automatically extract research data on bleeding by identifying bleeding events and their anatomical location in Danish unstructured EHR text. The pipeline consists of two deep learning classification models. The first model classifies if sentences indicate a bleeding event. The second model takes sentences with a bleeding event as input and classifies the location of the bleeding. The bleeding event classification model was trained by fine-tuning the Clin-ELECTRA LLM pre-trained on 1.4 billion words from 299,718 Danish EHRs for the task of bleeding classification. The model was fine-tuned on 25,862 labelled sentences and has a sensitivity and specificity of 95.8% on a test set balanced on bleeding indication and location. It has a sensitivity of 93.7% and specificity of 98.1% on an unbalanced test cohort of 566 admission. The model has sensitivities between 87.6% and 100% on different anatomical locations, and there are no major differences between minor and major bleeding, males and females, and young and elderly. The bleeding location classification model classifies the locations: CNS, eye, ENT, airway, GI, internal, gynaecological, urogenital, dermatological, and unknown. The model was trained by fine-tuning the Clin-ELECTRA LLM for bleeding location classification on 17,344 labelled sentences. The model has an overall accuracy of 90.5% on a balanced test set. It performs worst on internal bleeding at 78.67% and best on urogenital bleeding at 99.33%.

Chapter 5 addressed research question 2. I presented a deep learning pipeline to automatically extract clinical research data on VTE by identifying VTE events and their anatomical location in Danish unstructured EHR text. The pipeline consists of two deep learning classification models. The first model classifies if sentences indicate a VTE event. The second model takes sentences with a VTE event as input and classifies the location of the VTE. The VTE event classification model was trained by fine-tuning the MeDa-BERT LLM, which was initialised from Da-BERT and further adaptively pre-trained on 133 million tokens from a Danish medical corpus consisting of medical books and texts from the internet. For the VTE event model, MeDa-BERT was fine-tuned on 22,528 labelled samples, and it performs with 91.7% accuracy on a balanced test set. The VTE location classification model classifies the locations: lower extremity, lungs, and unknown. It was trained by fine-tuning MeDa-BERT on 6,000 labelled samples, and it performs with 95.8% accuracy on a balanced test set.

Chapter 6 addressed research question 3. I presented a system for automatic extraction of relevant parameters from the medical history in the form of clinical events, their attributes, and relations contained in Danish unstructured EHR text. This can aid in providing a structured overview of the medical history and support, e.g. scientific hypothesis generation for clinical research. The system was adapted from the PURE architecture for NER and RE. It uses Clin-ELECTRA as base and consists of three separate models to extract clinical events, attributes of the events, and relations between the events. Clinical events are: diseases, symptoms including abnormal findings, diagnostics, treatments, anatomies including body fluids and excrements, and results. Attributes are: prior, current, future, doubt, negation, and non-patient. Relations are: “has location”, “is treated with”, and “has result”. The dataset for training, validating, and testing the system contained 54,631 clinical events, 41,954 attributes, and 14,604 relations. The clinical event extraction model has a micro F1 of 63.22% (type-wise accuracy range 53%–69%), the attribute extraction model 66.04% (23%–84%), and the RE model 74.23% (62%–93%).

Chapter 7 addressed research question 4. I presented a deep learning model for automatic de-identification of names, streets, and locations in Danish unstructured EHR text. De-identification is necessary before data are used for research unless explicit consent is acquired. Using weakly supervised learning to develop the de-identification model did not require any manually labelled training data. The model is based on the PURE entity extraction architecture, using the Da-BERT LLM as base. The training data were labelled by a heuristic function that matched the words to lists of names, streets, and locations from official sources, and ambiguous identifiers were discarded from the training data if they occurred more in the corpus than expected for an identifier. The model de-identifies names, streets, and locations in unstructured EHR text with a sensitivity of 93.43%, precision of 86.10%, and F1 of 89.62%. It correctly de-identifies 91% of locations, and 94% of names and streets.

Chapter 8 addressed research question 5. I presented two Danish clinical word embeddings that can be used to create a dense clinical annotation corpus and increase the efficiency of the annotation process. I also presented a Danish clinical word embedding benchmark that can be used to ensure that the models perform well on clinical domain text. The benchmark consists of eight intrinsic tasks, evaluating inherent information on syntax and semantics, and two extrinsic tasks, evaluating performance in downstream classification tasks. The word embedding models can be used to achieve more efficient annotation of data, aiding in creating a dense annotation corpus. By having MDs create a list of words associated with, e.g. bleeding, and expanding that list using nearest neighbour calculation in word embedding space, the final list can be used to match words and thereby filter the corpus. The filtered dense corpus can achieve a more even proportion of samples in each class than an unfiltered one. This makes the annotation process more efficient.

9.1 Future Work

In the future, I would like to further apply the developed models to de-identify unstructured EHR text and extract parameters for clinical research. The bleeding event classification model has already been applied to one clinical research project, and the pipeline for identification of bleeding events and their location will be used for two more project as described in section 4.4.

The NER and RE models for clinical events, attributes, and relations can extract relevant information from the medical history, but the resulting data must be assessed by a human to be included as a parameter in clinical research because the models only classify broad classes as, e.g. "disease". The output of classification models like those developed for bleeding and VTE can be used directly as a parameter in research. In future work, I will develop more classification models to support clinical research by extracting other relevant parameters from the unstructured EHR text.

The models presented in this dissertation can all potentially improve their performance by re-training on more labelled data. In future work, I will continue to collect and annotate more data to improve the performance of the models. A special focus will be on further developing the VTE location classification model to also classify cerebral, eye, cardiac, liver, kidney, intestines, and upper extremity locations. Other future work with the VTE models consists of testing them with a Clin-ELECTRA base instead of a MeDa-BERT base. For the bleeding location classification model, more muscle and joint bleedings and sentences indicating multiple bleedings will be annotated. The method for achieving a more efficient annotation process, described in chapter 8, will be validated and used to create a dense corpus targeted at the specific classes lacking in samples.

Finally, I would like to explore prompt engineering to possibly improve de-identification and medical history extraction. The de-identification model currently de-identifies names, streets, and locations. In the future, de-identification of other identifiers like dates, phone numbers, and e-mail addresses must be added. The performance of the models for NER and RE on the medical history must also be improved. While this can most likely be achieved by training on more annotated data, it is interesting to explore the use of few-shot or zero-shot prompting of LLMs such as GPT-3, ChatGPT, and GPT-4 as this technique does not require any annotated data and has shown promising results [132].

Bibliography

[1] Paul C Tang and Clement J McDonald. "Electronic health record systems". In: *Biomedical informatics: computer applications in health care and biomedicine*, 2006, pp. 447–475.

[2] Eleni Entzeridou, Evgenia Markopoulou, and Vasiliki Mollaki. "Public and physician's expectations and ethical concerns about electronic health record: Benefits outweigh risks except for information security". In: *International journal of medical informatics* vol. 110, 2018, pp. 98–107.

[3] Michela Assale, Linda Greta Dui, Andrea Cina, Andrea Seveso, and Federico Cabitza. "The revival of the notes field: leveraging the unstructured content in electronic health records". In: *Frontiers in medicine* vol. 6, 2019, p. 66.

[4] Hyoun-Joong Kong. "Managing unstructured big data in healthcare system". In: *Healthcare informatics research* vol. 25, no. 1, 2019, pp. 1–2.

[5] Sundhedsdatastyrelsen. *Klassifikation af sygdomme*. 2021. URL: https://sundhedsdatastyrelsen.dk/da/rammer-og-retningslinjer/om-klassifikationer/sks-klassifikationer/klassifikation-sygdomme (visited on 08/26/2023).

[6] World Health Organization. *Importance of ICD*. URL: https://www.who.int/standards/classifications/frequently-asked-questions/importance-of-icd (visited on 08/26/2023).

[7] Sissel Toft Sørensen, Frederik Pagh Kristensen, Frederikke Schønfeldt Troelsen, Morten Schmidt, and Henrik Toft Sørensen. "Health registries as research tools: a review of methodological key issues". In: *Danish medical journal* vol. 70, no. 4, 2023, A12220796.

[8] Morten Schmidt, Sigrun Alba Johannesdottir Schmidt, Kasper Adelborg, Jens Sundbøll, Kristina Laugesen, Vera Ehrenstein, and Henrik Toft Sørensen. "The Danish health care system and epidemiological research: from health care contacts to database records". In: *Clinical epidemiology*, 2019, pp. 563–591.

[9] Indenrigs og Sundhedsministeriet. *Bekendtgørelse om lægers anmeldelse til Cancerregisteret af kræftsygdomme m.v.* 2018. URL: https://www.retsinformation.dk/eli/lta/2018/1049 (visited on 08/26/2023).

[10] Morten Schmidt, Sigrun Alba Johannesdottir Schmidt, Jakob Lynge Sandegaard, Vera Ehrenstein, Lars Pedersen, and Henrik Toft Sørensen. "The Danish National Patient Registry: a review of content, data quality, and research potential". In: *Clinical epidemiology*, 2015, pp. 449–490.

[11] Andrea J Darzi, Samer G Karam, Rana Charide, Itziar Etxeandia-Ikobaltzeta, Mary Cushman, Michael K Gould, Lawrence Mbuagbaw, Frederick A Spencer, Alex C Spyropoulos, Michael B Streiff, et al. "Prognostic factors for VTE and bleeding in hospitalized medical patients: a systematic review and meta-analysis". In: *Blood* vol. 135, no. 20, 2020, pp. 1788–1810.

[12] David A Grimes. "Epidemiologic research using administrative databases: garbage in, garbage out". In: *Obstetrics & Gynecology* vol. 116, no. 5, 2010, pp. 1018–1019.

[13] Milena A Gianfrancesco and Neal D Goldstein. "A narrative review on the validity of electronic health record-based research in epidemiology". In: *BMC Medical Research Methodology* vol. 21, no. 1, 2021, pp. 1–10.

[14] Kimberly J O'malley, Karon F Cook, Matt D Price, Kimberly Raiford Wildes, John F Hurdle, and Carol M Ashton. "Measuring diagnoses: ICD code accuracy". In: *Health services research* vol. 40, no. 5p2, 2005, pp. 1620–1639.

[15] John A Baron and Elisabete Weiderpass. "An introduction to epidemiological research with medical databases". In: *Annals of epidemiology* vol. 10, no. 4, 2000, pp. 200–204.

[16] Jeffrey S Berger, Deepak L Bhatt, P Gabriel Steg, Steven R Steinhubl, Gilles Montalescot, Mingyuan Shao, Werner Hacke, Keith A Fox, Peter B Berger, Eric J Topol, et al. "Bleeding, mortality, and antiplatelet therapy: results from the Clopidogrel for High Atherothrombotic Risk and Ischemic Stabilization, Management, and Avoidance (CHARISMA) trial". In: *American heart journal* vol. 162, no. 1, 2011, pp. 98–105.

[17] Deborah J Cook, Lauren E Griffith, Stephen D Walter, Gordon H Guyatt, Maureen O Meade, Daren K Heyland, Ann Kirby, Michael Tryba, and Canadian Critical Care Trials Group. "The attributable mortality and length of intensive care unit stay of clinically important gastrointestinal bleeding in critically ill patients". In: *Critical care* vol. 5, 2001, pp. 1–8.

[18] Hervé Decousus, Victor F Tapson, Jean-François Bergmann, Beng H Chong, James B Froehlich, Ajay K Kakkar, Geno J Merli, Manuel Monreal, Mashio Nakamura, Ricardo Pavanello, et al. "Factors at admission associated with bleeding risk in medical patients: findings from the IMPROVE investigators". In: *Chest* vol. 139, no. 1, 2011, pp. 69–79.

[19] Vera E Valkhoff, Preciosa M Coloma, Gwen MC Masclee, Rosa Gini, Francesco Innocenti, Francesco Lapi, Mariam Molokhia, Mees Mosseveld, Malene Schou Nielsson, Martijn Schuemie, et al. "Validation study in four health-care databases: upper gastrointestinal bleeding misclassification affects precision but not magnitude of drug-related upper gastrointestinal bleeding risk". In: *Journal of clinical epidemiology* vol. 67, no. 8, 2014, pp. 921–931.

[20] Lise R Øie, Mattis A Madsbu, Charalampis Giannadakis, Anders Vorhaug, Heidi Jensberg, Øyvind Salvesen, and Sasha Gulati. "Validation of intracranial hemorrhage in the Norwegian Patient Registry". In: *Brain and behavior* vol. 8, no. 2, 2018, e00900.

[21] Thomas Delate, Aubrey E Jones, Nathan P Clark, and Daniel M Witt. "Assessment of the coding accuracy of warfarin-related bleeding events". In: *Thrombosis research* vol. 159, 2017, pp. 86–90.

[22] Susan E Andrade, Jerry H Gurwitz, K Arnold Chan, James G Donahue, Arne Beck, Myde Boles, Diana SM Buist, Michael Goodman, Andrea Z LaCroix, TR Levin, et al. "Validation of diagnoses of peptic ulcers and bleeding from administrative databases: a multi-health maintenance organization study". In: *Journal of clinical epidemiology* vol. 55, no. 3, 2002, pp. 310–313.

[23] Meng-Tsang Hsieh, Kuo-Chang Huang, Cheng-Yang Hsieh, Tzu-Tung Tsai, Li-Ching Chen, and Sheng-Feng Sung. "Validation of ICD-10-CM diagnosis codes for identification of patients with acute hemorrhagic stroke in a national health insurance claims database". In: *Clinical Epidemiology*, 2021, pp. 43–51.

[24] Natalie McCormick, Vidula Bhole, Diane Lacaille, and J Antonio Avina-Zubieta. "Validity of diagnostic codes for acute stroke in administrative databases: a systematic review". In: *PloS one* vol. 10, no. 8, 2015, e0135834.

[25] Samantha J Lain, Christine L Roberts, Ruth M Hadfield, Jane C Bell, and Jonathan M Morris. "How accurate is the reporting of obstetric haemorrhage in hospital discharge data? A validation study". In: *Australian and New Zealand Journal of Obstetrics and Gynaecology* vol. 48, no. 5, 2008, pp. 481–484.

[26] T Arnason, PS Wells, C Van Walraven, and AJ Forster. "Accuracy of coding for possible warfarin complications in hospital discharge abstracts". In: *Thrombosis research* vol. 118, no. 2, 2006, pp. 253–262.

[27] Christopher Joos, Kevin Lawrence, Aubrey E Jones, Stacy A Johnson, and Daniel M Witt. "Accuracy of ICD-10 codes for identifying hospitalizations for acute anticoagulation therapy-related bleeding events". In: *Thrombosis Research* vol. 181, 2019, pp. 71–76.

[28] National Heart Lung and Blood Institute. *What Is Venous Thromboembolism?* 2022. URL: `https://www.nhlbi.nih.gov/health/venous-thromboembolism` (visited on 08/26/2023).

[29] Alain Leizorovicz, Alexander T Cohen, Alexander GG Turpie, Carl-Gustav Olsson, Paul T Vaitkus, and Samuel Z Goldhaber. "Randomized, placebo-controlled trial of dalteparin for the prevention of venous thromboembolism in acutely ill medical patients". In: *Circulation* vol. 110, no. 7, 2004, pp. 874–879.

[30] Meyer Michel Samama, Alexander Thomas Cohen, Jean-Yves Darmon, Louis Desjardins, Amiram Eldor, Charles Janbon, Alain Leizorovicz, Hélène Nguyen, Carl-Gustav Olsson, Alexander Graham Turpie, et al. "A comparison of enoxaparin with placebo for the prevention of venous thromboembolism in acutely ill medical patients". In: *New England Journal of Medicine* vol. 341, no. 11, 1999, pp. 793–800.

[31] John A Heit, L Joseph Melton III, Christine M Lohse, Tanya M Petterson, Marc D Silverstein, David N Mohr, and W Michael O'fallon. "Incidence of venous thromboembolism in hospitalized patients vs community residents". In: *Mayo Clinic Proceedings*. Vol. 76. 11. Elsevier. 2001, pp. 1102–1110.

[32] Alexander T Cohen, Giancarlo Agnelli, Frederick A Anderson, Juan I Arcelus, David Bergqvist, Josef G Brecht, Ian A Greer, John A Heit, Julia L Hutchinson, Ajay K Kakkar, et al. "Venous thromboembolism (VTE) in Europe". In: *Thrombosis and haemostasis* vol. 98, no. 10, 2007, pp. 756–764.

[33] Kirstine Kobberøe Søgaard, Morten Schmidt, Lars Pedersen, Erzsébet Horváth-Puhó, and Henrik Toft Sørensen. "30-year mortality after venous thromboembolism: a population-based cohort study". In: *Circulation* vol. 130, no. 10, 2014, pp. 829–836.

[34] Marianne Tang Severinsen, Søren Risom Kristensen, Kim Overvad, Claus Dethlefsen, Anne Tjønneland, and Søren Paaske Johnsen. "Venous thromboembolism discharge diagnoses in the Danish National Patient Registry should be used with caution". In: *Journal of clinical epidemiology* vol. 63, no. 2, 2010, pp. 223–228.

[35] Richard H White, Leslea A Brickner, and Kate A Scannell. "ICD-9-CM codes poorly indentified venous thromboembolism during pregnancy". In: *Journal of clinical epidemiology* vol. 57, no. 9, 2004, pp. 985–988.

[36] Christine Baumgartner, Alan S Go, Dongjie Fan, Sue Hee Sung, Daniel M Witt, John R Schmelzer, Marc S Williams, Steven H Yale, Jeffrey J VanWormer, and Margaret C Fang. "Administrative codes inaccurately identify recurrent venous thromboembolism: the CVRN VTE study". In: *Thrombosis research* vol. 189, 2020, pp. 112–118.

[37] Pierre Casez, José Labarère, Marie-Antoinette Sevestre, Myriam Haddouche, Xavier Courtois, Sandrine Mercier, Elisabeth Lewandowski, Jérôme Fauconnier, Patrice François, and Jean-Luc Bosson. "ICD-10 hospital discharge diagnosis codes were sensitive for identifying pulmonary embolism but not deep vein thrombosis". In: *Journal of clinical epidemiology* vol. 63, no. 7, 2010, pp. 790–797.

[38] Margaret C Fang, Dongjie Fan, Sue Hee Sung, Daniel M Witt, John R Schmelzer, Steven R Steinhubl, Steven H Yale, and Alan S Go. "Validity of using inpatient and outpatient administrative codes to identify acute venous thromboembolism: the CVRN VTE study". In: *Medical care* vol. 55, no. 12, 2017, e137.

[39] Brian R Branchford, Elizabeth Gibson, Marilyn J Manco-Johnson, and Neil A Goldenberg. "Sensitivity of discharge diagnosis ICD-9 codes for pediatric venous thromboembolism is greater than specificity, but still suboptimal for surveillance and clinical research". In: *Thrombosis research* vol. 129, no. 5, 2012, pp. 662–663.

[40] The European Parliament and The Council Of The European Union. *General Data Protection Regulation*. 2018. URL: `https://gdpr-info.eu/` (visited on 08/26/2023).

[41] Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. "Analyzing Leakage of Personally Identifiable Information in Language Models". In: *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society. 2023, pp. 346–363.

[42] Robert Leaman, Ritu Khare, and Zhiyong Lu. "Challenges in clinical natural language processing for automated disorder normalization". In: *Journal of biomedical informatics* vol. 57, 2015, pp. 28–37.

[43] Georg Rehm, Maria Berger, Ela Elsholz, Stefanie Hegele, Florian Kintzel, Katrin Marheinecke, Stelios Piperidis, Miltos Deligiannis, Dimitrios Galanis, Katerina Gkirtzou, et al. "European Language Grid: An Overview". In: *Proceedings of the Twelfth Language Resources and Evaluation Conference*. 2020, pp. 3366–3380.

[44] Binggui Zhou, Guanghua Yang, Zheng Shi, and Shaodan Ma. "Natural language processing for smart healthcare". In: *IEEE Reviews in Biomedical Engineering*, 2022.

[45] Paroma Varma and Christopher Ré. "Snuba: Automating weak supervision to label training data". In: *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*. Vol. 12. 3. NIH Public Access. 2018, p. 223.

[46] Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Rasmus Søgaard Hansen, Thiusius Rajeeth Savarimuthu, Rasmus Bank Lynggaard, and Pernille Just Vinholt. "Doctors identify hemorrhage better during chart review when assisted by artificial intelligence". In: *Applied Clinical Informatics*, 2023.

[47] Jannik S Pedersen, Martin S Laursen, Cristina Soguero-Ruiz, Thiusius R Savarimuthu, Rasmus Søgaard Hansen, and Pernille J Vinholt. "Domain over size: Clinical ELECTRA surpasses general BERT for bleeding site classification in the free text of electronic health

records". In: *2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE. 2022, pp. 1–4.

[48] Martin Laursen, Jannik Pedersen, Rasmus Hansen, Thiusius Rajeeth Savarimuthu, and Pernille Vinholt. "Danish Clinical Named Entity Recognition and Relation Extraction". In: *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 655–666.

[49] Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Vinholt, and Thiusius R Savarimuthu. "Automatic Annotation of Training Data for Deep Learning Based De-identification of Narrative Clinical Text". In: *WNLPe-Health 2022: Proceedings of The First Workshop on Context-aware NLP in eHealth (WNLPe-Health 2022)*. CEUR Workshop Proceedings. 2023, pp. 30–44.

[50] Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Just Vinholt, Rasmus Søgaard Hansen, and Thiusius Rajeeth Savarimuthu. "Benchmark for evaluation of danish clinical word embeddings". In: *Northern European Journal of Language Technology* vol. 9, no. 1, 2023.

[51] Jannik Pedersen, Martin Laursen, Pernille Vinholt, and Thiusius Rajeeth Savarimuthu. "MeDa-BERT: A medical Danish pretrained transformer model". In: *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 301–307.

[52] Jannik S Pedersen, Martin S Laursen, Thiusius Rajeeth Savarimuthu, Rasmus Søgaard Hansen, Anne Bryde Alnor, Kristian Voss Bjerre, Ina Mathilde Kjær, Charlotte Gils, Anne-Sofie Faarvang Thorsen, Eline Sandvig Andersen, Søren Andreas Just, and Pernille Just Vinholt. "Deep learning detects and visualizes bleeding events in electronic health records". In: *Research and practice in thrombosis and haemostasis* vol. 5, no. 4, 2021, e12505.

[53] Martin S Laursen, Jannik S Pedersen, Søren A Just, Thiusius R Savarimuthu, Brian Blomholt, Jakob KH Andersen, and Pernille J Vinholt. "Factors Facilitating the Acceptance of Diagnostic Robots in Healthcare: A Survey". In: *2022 IEEE 10th International Conference on Healthcare Informatics (ICHI)*. IEEE. 2022, pp. 442–448.

[54] Jannik Pedersen, Martin Laursen, Pernille Vinholt, Anne Alnor, and Thiusius Savarimuthu. "Investigating anatomical bias in clinical machine learning algorithms". In: *Findings of the Association for Computational Linguistics: EACL 2023*. 2023, pp. 1368–1380.

[55] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. "Natural language processing: an introduction". In: *Journal of the American Medical Informatics Association* vol. 18, no. 5, 2011, pp. 544–551.

[56] Prashant Johri, Sunil K Khatri, Ahmad T Al-Taani, Munish Sabharwal, Shakhzod Suvanov, and Avneesh Kumar. "Natural language processing: History, evolution, application, and future work". In: *Proceedings of 3rd International Conference on Computing Informatics and Networks: ICCIN 2020*. Springer. 2021, pp. 365–375.

[57] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding". In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. 2018, pp. 353–355.

[58] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. "Superglue: A stickier benchmark for general-purpose language understanding systems". In: *Advances in neural information processing systems* vol. 32, 2019.

[59] Jeffrey L Elman. "Finding structure in time". In: *Cognitive science* vol. 14, no. 2, 1990, pp. 179–211.

[60] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. "Recurrent neural network based language model". In: *INTERSPEECH-2010*. ISCA. 2010, pp. 1045–1048.

[61] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* vol. 30, 2017.

[62] Milad Mohammadi, Rohit Mundra, Richard Socher, and Lisa Wang. *CS224N: Natural Language Processing with deep learning*. 2018. URL: `https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/lecture_notes/cs224n-2017-notes5.pdf`.

[63] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. "A Neural Probabilistic Language Model". In: *Journal of Machine Learning Research* vol. 3, 2003, pp. 1137–1155.

[64] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient estimation of word representations in vector space". In: *arXiv preprint arXiv:1301.3781*, 2013.

[65] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. "Enriching word vectors with subword information". In: *Transactions of the association for computational linguistics* vol. 5, 2017, pp. 135–146.

[66] Jeffrey Pennington, Richard Socher, and Christopher Manning. "GloVe: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. DOI: `10.3115/v1/D14-1162`. URL: `https://aclanthology.org/D14-1162`.

[67] Priyankar Bose, Sriram Srinivasan, William C Sleeman IV, Jatinder Palta, Rishabh Kapoor, and Preetam Ghosh. "A survey on recent named entity recognition and relationship extraction techniques on clinical texts". In: *Applied Sciences* vol. 11, no. 18, 2021, p. 8319.

[68] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems* vol. 27, 2014.

[69] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* vol. 9, no. 8, 1997, pp. 1735–1780.

[70] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: `http://arxiv.org/abs/1409.0473`.

[71] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. DOI: `10.3115/v1/D14-1179`. URL: `https://aclanthology.org/D14-1179`.

[72] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* vol. 86, no. 11, 1998, pp. 2278–2324.

[73] Ronan Collobert and Jason Weston. "A unified architecture for natural language processing: Deep neural networks with multitask learning". In: *Proceedings of the 25th international conference on Machine learning*. 2008, pp. 160–167.

[74] Ye Zhang and Byron C Wallace. "A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2017, pp. 253–263.

[75] Alon Jacovi, Oren Sar Shalom, and Yoav Goldberg. "Understanding Convolutional Neural Networks for Text Classification". In: *Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics (ACL). 2018.

[76] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. "Language models are few-shot learners". In: *Advances in neural information processing systems* vol. 33, 2020, pp. 1877–1901.

[77] OpenAI. *GPT-4 Technical Report*. 2023. arXiv: `2303.08774 [cs.CL]`.

[78] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: `10.18653/v1/N19-1423`. URL: `https://aclanthology.org/N19-1423`.

[79] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. "ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators". In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL: `https://openreview.net/forum?id=r1xMH1BtvB`.

[80] Olga Petrova. *Natural language processing: The age of transformers*. 2020. URL: `https://towardsdatascience.com/natural-language-processing-the-age-of-transformers-a36c0265937d`.

[81] Kheirie Elhariri. *The Transformer model*. 2022. URL: `https://towardsdatascience.com/attention-is-all-you-need-e498378552f9`.

[82] Jay Alammar. *The illustrated transformer*. 2018. URL: `https://jalammar.github.io/illustrated-transformer/`.

[83] Sasha Targ, Diogo Almeida, and Kevin Lyman. "Resnet in resnet: Generalizing residual architectures". In: *arXiv preprint arXiv:1603.08029*, 2016.

[84] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. "Layer normalization". In: *arXiv preprint arXiv:1607.06450*, 2016.

[85] Maxwell Taggart, Wendy W Chapman, Benjamin A Steinberg, Shane Ruckel, Arianna Pregenzer-Wenzler, Yishuai Du, Jeffrey Ferraro, Brian T Bucher, Donald M Lloyd-Jones, Matthew T Rondina, et al. "Comparison of 2 natural language processing methods for identification of bleeding among critically ill patients". In: *JAMA network open* vol. 1, no. 6, 2018, e183451–e183451.

[86] Hee-Jin Lee, Min Jiang, Yonghui Wu, Christian M Shaffer, John H Cleator, Eitan A Friedman, Joshua P Lewis, Dan M Roden, Josh Denny, and Hua Xu. "A comparative study of different methods for automatic identification of clopidogrel-induced bleedings in electronic health records". In: *AMIA Summits on Translational Science Proceedings* vol. 2017, 2017, p. 185.

[87] Kamal Jnawali, Mohammad R Arbabshirani, Alvaro E Ulloa, Navalgund Rao, and Aalpen A Patel. "Automatic classification of radiological report for intracranial hemorrhage". In: *2019 IEEE 13th international conference on semantic computing (ICSC)*. IEEE. 2019, pp. 187–190.

[88] Rumeng Li, Baotian Hu, Feifan Liu, Weisong Liu, Francesca Cunningham, David D McManus, Hong Yu, et al. "Detection of bleeding events in electronic health record notes using convolutional neural network models enhanced with recurrent neural network autoencoders: deep learning approach". In: *JMIR medical informatics* vol. 7, no. 1, 2019, e10788.

[89] Avijit Mitra, Bhanu Pratap Singh Rawat, David McManus, Alok Kapoor, and Hong Yu. "Bleeding entity recognition in electronic health records: a comprehensive analysis of end-to-end systems". In: *AMIA Annual Symposium Proceedings*. Vol. 2020. American Medical Informatics Association. 2020, p. 860.

[90] Zhe Tian, Simon Sun, Tewodros Eguale, and Christian M Rochefort. "Automated extraction of VTE events from narrative radiology reports in electronic health records: a validation study". In: *Medical care* vol. 55, no. 10, 2017, e73.

[91] Jorge A Gálvez, Janine M Pappas, Luis Ahumada, John N Martin, Allan F Simpao, Mohamed A Rehman, and Char Witmer. "The use of natural language processing on pediatric diagnostic radiology reports in the electronic health record to identify deep venous thrombosis in children". In: *Journal of Thrombosis and Thrombolysis* vol. 44, 2017, pp. 281–290.

[92] Raymund B Dantes, Shuai Zheng, James J Lu, Michele G Beckman, Asha Krishnaswamy, Lisa C Richardson, Sheri Chernetsky-Tejedor, and Fusheng Wang. "Improved identification of venous thromboembolism from electronic medical records using a novel information extraction software platform". In: *Medical care* vol. 56, no. 9, 2018, e54.

[93] Romil F Shah, Stefano Bini, and Thomas Vail. "Data for registry and quality review can be retrospectively collected using natural language processing from unstructured charts of arthroplasty patients". In: *The Bone & Joint Journal* vol. 102, no. 7 Supple B, 2020, pp. 99–104.

[94] Stacy A Johnson, Emily A Signor, Katie L Lappe, Jianlin Shi, Stephen L Jenkins, Sara W Wikstrom, Rachel D Kroencke, David Hallowell, Aubrey E Jones, and Daniel M Witt. "A comparison of natural language processing to ICD-10 codes for identification and characterization of pulmonary embolism". In: *Thrombosis Research* vol. 203, 2021, pp. 190–195.

[95] Amol A Verma, Hassan Masoom, Chloe Pou-Prom, Saeha Shin, Michael Guerzhoy, Michael Fralick, Muhammad Mamdani, and Fahad Razak. "Developing and validating natural language processing algorithms for radiology reports compared to ICD-10 codes for identifying venous thromboembolism in hospitalized medical patients". In: *Thrombosis Research* vol. 209, 2022, pp. 51–58.

[96] Ang Li, Wilson L da Costa Jr, Danielle Guffey, Emily M Milner, Anthony K Allam, Karen M Kurian, Francisco J Novoa, Marguerite D Poche, Raka Bandyo, Carolina Granada,

et al. “Developing and optimizing a computable phenotype for incident venous thromboembolism in a longitudinal cohort of patients with cancer”. In: *Research and practice in thrombosis and haemostasis* vol. 6, no. 4, 2022, e12733.

[97] Zhi-Geng Jin, Hui Zhang, Mei-Hui Tai, Ying Yang, Yuan Yao, and Yu-Tao Guo. “Natural Language Processing in a Clinical Decision Support System for the Identification of Venous Thromboembolism: Algorithm Development and Validation”. In: *Journal of Medical Internet Research* vol. 25, 2023, e43153.

[98] Christian M Rochefort, Aman D Verma, Tewodros Eguale, Todd C Lee, and David L Buckeridge. “A novel method of adverse event detection can accurately identify venous thromboembolisms (VTEs) from narrative electronic health record data”. In: *Journal of the American Medical Informatics Association* vol. 22, no. 1, 2015, pp. 155–165.

[99] Ivan Lerner, Nicolas Paris, and Xavier Tannier. “Terminologies augmented recurrent neural network model for clinical named entity recognition”. In: *Journal of biomedical informatics* vol. 102, 2020, p. 103356.

[100] Suyu Ge, Fangzhao Wu, Chuhan Wu, Tao Qi, Yongfeng Huang, and Xing Xie. “Fedner: Privacy-preserving medical named entity recognition with federated learning”. In: *arXiv preprint arXiv:2003.09288*, 2020.

[101] Xin Yu, Wenshen Hu, Sha Lu, Xiaoyan Sun, and Zhenming Yuan. “BioBERT based named entity recognition in electronic medical record”. In: *2019 10th international conference on information technology in medicine and education (ITME)*. IEEE. 2019, pp. 49–52.

[102] Fenia Christopoulou, Thy Thy Tran, Sunil Kumar Sahu, Makoto Miwa, and Sophia Ananiadou. “Adverse drug events and medication relation extraction in electronic health records with ensemble deep learning methods”. In: *Journal of the American Medical Informatics Association* vol. 27, no. 1, 2020, pp. 39–46.

[103] Elisa Terumi Rubel Schneider, João Vitor Andrioli de Souza, Julien Knafou, Lucas Emanuel Silva e Oliveira, Jenny Copara, Yohan Bonescki Gumiel, Lucas Ferro Antunes de Oliveira, Emerson Cabrera Paraiso, Douglas Teodoro, and Cláudia Maria Cabral Moro Barra. “BioBERTpt-a Portuguese neural language model for clinical named entity recognition”. In: *Proceedings of the 3rd Clinical Natural Language Processing Workshop*. 2020, pp. 65–72.

[104] Kostas Pantazos, Soren Lauesen, and Soren Lippert. “Preserving medical correctness, readability and consistency in de-identified health records”. In: *Health informatics journal* vol. 23, no. 4, 2017, pp. 291–303.

[105] Hanna Berg and Hercules Dalianis. “A Semi-supervised Approach for De-identification of Swedish Clinical Text”. In: *Proceedings of the Twelfth Language Resources and Evaluation Conference*. 2020, pp. 4444–4450.

[106] Zengjian Liu, Buzhou Tang, Xiaolong Wang, and Qingcai Chen. “De-identification of clinical notes via recurrent neural network and conditional random field”. In: *Journal of biomedical informatics* vol. 75, 2017, S34–S42.

[107] U.S. Department of Health and Human Services. *Health Insurance Portability and Accountability Act of 1996 (HIPAA), Public Law 104-191*. 1996. URL: `https://www.hhs.gov/hipaa/for-professionals/index.html` (visited on 08/26/2023).

[108] Rosario Catelli, Francesco Gargiulo, Valentina Casola, Giuseppe De Pietro, Hamido Fujita, and Massimo Esposito. “A novel covid-19 data set and an effective deep learning approach for the de-identification of italian medical records”. In: *Ieee Access* vol. 9, 2021, pp. 19097–19110.

[109] Buzhou Tang, Dehuan Jiang, Qingcai Chen, Xiaolong Wang, Jun Yan, and Ying Shen. "De-identification of clinical text via Bi-LSTM-CRF with neural language models". In: *AMIA Annual Symposium Proceedings*. Vol. 2019. American Medical Informatics Association. 2019, p. 857.

[110] Rosario Catelli, Francesco Gargiulo, Emanuele Damiano, Massimo Esposito, and Giuseppe De Pietro. "Clinical de-identification using sub-document analysis and ELECTRA". in: *2021 IEEE International Conference on Digital Health (ICDH)*. IEEE. 2021, pp. 266–275.

[111] Zhengliang Liu, Xiaowei Yu, Lu Zhang, Zihao Wu, Chao Cao, Haixing Dai, Lin Zhao, Wei Liu, Dinggang Shen, Quanzheng Li, et al. "Deid-gpt: Zero-shot medical text de-identification by gpt-4". In: *arXiv preprint arXiv:2303.11032*, 2023.

[112] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation". In: *arXiv preprint arXiv:1609.08144*, 2016.

[113] Hugging Face. *ELECTRA*. URL: `https://huggingface.co/docs/transformers/model_doc/electra` (visited on 08/26/2023).

[114] Rasmus Søgaard Hansen. "Elucidating bleeding episodes and the clinical impact of bleeding". PhD thesis. University of Southern Denmark, 2023.

[115] Alex C Spyropoulos, Frederick A Anderson Jr, Gordon FitzGerald, Herve Decousus, Mario Pini, Beng H Chong, Rainer B Zotz, Jean-François Bergmann, Victor Tapson, James B Froehlich, et al. "Predictive and associative models to identify hospitalized medical patients at risk for VTE". in: *Chest* vol. 140, no. 3, 2011, pp. 706–714.

[116] Certainly. *Nordic BERT*. URL: `https://github.com/certainlyio/nordic_bert` (visited on 08/26/2023).

[117] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 8342–8360.

[118] MA Oquendo, E Baca-Garcia, A Artes-Rodriguez, F Perez-Cruz, HC Galfalvy, H Blasco-Fontecilla, D Madigan, and N Duan. "Machine learning and data mining: strategies for hypothesis generation". In: *Molecular psychiatry* vol. 17, no. 10, 2012, pp. 956–959.

[119] Xia Jing, Vimla L Patel, James J Cimino, Jay H Shubrook, Yuchun Zhou, Chang Liu, Sonsoles De Lacalle, et al. "The roles of a secondary data analytics tool and experience in scientific hypothesis generation in clinical research: protocol for a mixed methods study". In: *JMIR Research Protocols* vol. 11, no. 7, 2022, e39414.

[120] Xia Jing, Vimla L Patel, James J Cimino, Jay H Shubrook, Yuchun Zhou, Brooke N Draghi, Mytchell A Ernst, Chang Liu, and Sonsoles De Lacalle. "A Visual Analytic Tool (VIADS) to Assist the Hypothesis Generation Process in Clinical Research: Mixed Methods Usability Study". In: *JMIR Human Factors* vol. 10, 2023, e44644.

[121] Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. "A survey on deep learning for named entity recognition". In: *IEEE Transactions on Knowledge and Data Engineering* vol. 34, no. 1, 2020, pp. 50–70.

[122] Zexuan Zhong and Danqi Chen. "A Frustratingly Easy Approach for Entity and Relation Extraction". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2021, pp. 50–61.

[123] Xing Liu, Huiqin Chen, and Wangui Xia. “Overview of named entity recognition”. In: *Journal of Contemporary Educational Research* vol. 6, no. 5, 2022, pp. 65–68.

[124] Alexander Ratner, Stephen H Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. “Snorkel: Rapid training data creation with weak supervision”. In: *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*. Vol. 11. 3. NIH Public Access. 2017, p. 269.

[125] Jason A Fries, Ethan Steinberg, Saelig Khattar, Scott L Fleming, Jose Posada, Alison Callahan, and Nigam H Shah. “Ontology-driven weak supervision for clinical entity classification in electronic health records”. In: *Nature communications* vol. 12, no. 1, 2021, p. 2017.

[126] Zhe Jian, Xusheng Guo, Shijian Liu, Handong Ma, Shaodian Zhang, Rui Zhang, and Jianbo Lei. “A cascaded approach for Chinese clinical text de-identification with less annotation effort”. In: *Journal of biomedical informatics* vol. 73, 2017, pp. 76–83.

[127] Felipe Almeida and Geraldo Xexéo. “Word embeddings: A survey”. In: *arXiv preprint arXiv:1901.09069*, 2019.

[128] Zellig S Harris. “Distributional structure”. In: *Word* vol. 10, no. 2-3, 1954, pp. 146–162.

[129] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. “Distributed representations of words and phrases and their compositionality”. In: *Advances in neural information processing systems* vol. 26, 2013.

[130] Khalid Al Aboud. “Homonyms in medicine; a perspective”. In: *Our Dermatology Online* vol. 6, no. 1, 2015, p. 109.

[131] Lilian Weng. *Learning word embedding*. 2017. URL: `https://lilianweng.github.io/posts/2017-10-15-word-embedding/`.

[132] Jiaqi Wang, Enze Shi, Sigang Yu, Zihao Wu, Chong Ma, Haixing Dai, Qiushi Yang, Yanqing Kang, Jinru Wu, Huawen Hu, et al. “Prompt engineering for healthcare: Methodologies and applications”. In: *arXiv preprint arXiv:2304.14670*, 2023.

APPENDIX A

MeDa-BERT: A medical Danish pretrained transformer model

This is a reprint of our original paper [51], which is licensed under a CC BY 4.0 license. The CC BY 4.0 license can be found at https://creativecommons.org/licenses/by/4.0/.

Notice that, throughout the paper, everywhere that reads "airways" should be "lower extremity" instead.

MeDa-BERT: A medical Danish pretrained transformer model

Jannik Skyttegaard Pedersen*
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
jasp@mmmi.sdu.dk

Martin Sundahl Laursen*
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
msla@mmmi.sdu.dk

Pernille Just Vinholt
Department of Clinical Biochemistry
Odense University Hospital

Thiusius Rajeeth Savarimuthu
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Abstract

This paper introduces a medical Danish BERT-based language model (MeDa-BERT) and medical Danish word embeddings. The word embeddings and MeDa-BERT were pretrained on a new medical Danish corpus consisting of 133M tokens from medical Danish books and text from the internet. The models showed improved performance over general-domain models on medical Danish classification tasks. The medical word embeddings[1] and MeDa-BERT[2] are publicly available.

1 Introductions

Large language models (LLM) are powerful representation learners and have become the backbone structure of many modern natural language processing (NLP) systems. To learn text representations, LLM are first pretrained on a large-scale text corpus using self-supervised learning, e.g., masked language modelling. After pretraining, LLM are fine-tuned on specific downstream tasks where they have achieved state-of-the-art results on NLP benchmarks such as GLUE (Wang et al., 2018).

However, directly applying these general pretrained models to specialized domains such as the medical have led to unsatisfactory results (Peng et al., 2019). As a solution to this, a second round of in-domain pretraining (domain-adaptive pretraining) has shown to improve the performance of LLMs that were first trained on a general domain corpus (Gururangan et al., 2020). Domain-adaptive pretraining adjusts the weights of the LLM to better capture the terminology, style, and nuances that are relevant to the target domain.

Resource-rich languages such as English have large domain-specific corpuses available that have been used to develop e.g., biomedical (Lee et al., 2020), clinical (Alsentzer et al., 2019), scientific (Beltagy et al., 2019), and financial (Peng et al., 2021) LLMs that perform better than models trained on general corpuses. These models could potentially be used to improve human decision making, save time, and reduce costs, e.g., by extracting information from scientific articles, identifying potential drug interactions, and helping with NLP tasks such as text classification, named entity recognition, and question answering for each of their specialized domains.

For the Danish language, only LLMs trained on a general domain have been made publicly available[3]. This paper presents a medical Danish BERT model (MeDa-BERT)—a LLM trained on a new medical Danish text corpus. We also used the medical corpus to train medical word embeddings as they still have value in the clinical domain (Laursen et al., 2023). To evaluate the medical word embeddings and MeDa-BERT, we used existing medical Danish classification datasets. We found that an LSTM model using the medical word embeddings outperformed a similar model using general-domain word embeddings, and that MeDa-BERT performed slightly better than a general-domain BERT model.

2 Method

This section first describes how the medical corpus was collected and used to pretrain the medical Danish word embeddings and MeDa-BERT. Next, the datasets used to compare model performances and the fine-tuning procedure is described.

*Equal contribution
[1] https://huggingface.co/jannikskytt/MeDa-WE
[2] https://huggingface.co/jannikskytt/MeDa-Bert
[3] Pedersen et al. (2022b) developed a clinical transformer model but it is not publicly available

Corpus	Type	Date retrieved	Tokens
Clinical guidelines	Guidelines	October - November 2022	80,567,576
Medicin.dk	Information portal	June 2021	28,878,335
FADL	Books	January 2022	12,531,373
Sundhed.dk	Information portal	May 2022	6,767,409
Netdoktor.dk	Information portal	October 2022	3,227,051
Wikipedia	Encyclopedia	October 2022	1,992,796
Total			**133,964,540**

Table 1: Number of tokens and date retrieved for each data source

2.1 Danish medical corpus

We collected data from the internet and from medical books. The owners of the data resources approved that we used their data in this study. We describe the data collection for each text contributor below. An overview of the text corpuses and their size can be seen in Table 1.

2.1.1 Clinical guidelines

We collected text from the document management systems of the five Danish regions. The documents contain guidelines and instructions for diagnostics and treatment of patients and all workflows that support this. The document systems also include non-medical documents from e.g. purchasing, logistics, and service departments which were removed. All departments that were excluded and the number of tokens retrieved from each region can be seen in Appendix A.

2.1.2 Medical information portals

We collected text from webpages that provide information to medical doctors and patients. The text was collected from Medicin.dk, Netdoktor.dk, and Sundhed.dk. The resources provide information about diseases, symptoms, and medical treatments. Moreover, the resources contain information specifically for health care professionals, e.g., medication guidelines and information about best practices in the field. Text not related to the medical domain and text written by non-professionals were removed from the corpus. A description of this process can be seen in appendix A.

2.1.3 Books

This part of the corpus consisted of 107 medical books from publisher FADLs Forlag that publishes books for medicine and nursing school.

2.1.4 Wikipedia

We used PetScan[4] to search for medical Wikipedia documents within predefined categories and its subcategories. We used a maximum depth of 5 for searching for subcategories. The following categories were used: anatomi, physiology, diseases, medication, epidemiology, diagnostics, medical procedures, medical specialities, medical physics, and medical equipment. We excluded documents with the categories: persons and companies. This process resulted in 5,391 documents. Next, we manually removed non-medical articles from that list which resulted in 5,266 documents.

2.2 Preprocessing of data

For all text corpusses, we defined a sample as one paragraph, i.e., a continuous stream of text without line breaks. We inserted spaces between alphanumeric and non-alphanumeric characters. Samples were further preprocessed to fit the pretraining procedure for either word embeddings or the transformer model, as detailed below.

2.2.1 Danish medical transformer model

MeDa-BERT was initialized with weights from a pretrained Danish BERT model[5] trained on 10.7 GB Danish text from Common Crawl (9.5 GB), Danish Wikipedia (221 MB), debate forums (168 MB), and Danish OpenSubtitles (881 MB).

For domain-adaptive pretraining, samples from the collected medical corpus were appended a [CLS] and [SEP] token in the start and end of each sample, respectively. Samples were concatenated to fit the maximum sequence length of 512 tokens and document boundaries were indicated by adding an extra [SEP] token in between samples. After this process, we removed duplicates corresponding to 0.2% of the total corpus. The model was trained using Adam (Kingma and Ba, 2015) with a weight decay of 0.01 as described in (Loshchilov and Hutter). Using gradient accumulation, the model was trained with a batch size of 4,032, a learning rate of 1e-4, and a linear learning rate decay warmed up over 1 epoch. The model was pretrained for a total of 48 epochs and evaluated after 16, 32, and 48 epochs. We used 5% of the samples as a validation set to evaluate the model during pretraining and trained the model on the remaining data using dynamic

[4] `https://petscan.wmflabs.org/`
[5] `https://github.com/certainlyio/nordic_bert`

Dataset	Label	Train	Validation	Test
Bleeding	Positive	10,331	1,300	1,300
	Negative	10,331	1,300	1,300
Bleeding site	Airways	1,000	125	125
	Cerebral	1,000	125	125
	Ear-nose-throat	1,000	125	125
	Eyes	1,000	125	125
	Gastrointestinal	1,000	125	125
	Gynecological	1,000	125	125
	Internal	1,000	125	125
	Skin	1,000	125	125
	Urogenital	1,000	125	125
	Unknown	1,000	125	125
VTE	Positive	9,064	1,100	1,100
	Negative	9,064	1,100	1,100
VTE site	Airways	1,600	200	200
	Lungs	1,600	200	200
	Unknown	1,600	200	200

Table 2: Dataset distributions

masked language modeling. The model was optimized using four Tesla v100 GPUs using the Huggingface (Wolf et al., 2020) library. All model parameters and pretraining losses are shown in Appendix B.

2.3 Danish medical word embeddings

We trained 300-dimensional FastText (Bojanowski et al., 2017) word embeddings. The embeddings were trained for 10 epochs using a window size of 5 and 10 negative samples. The hyperparameters were chosen to be able to compare the produced embeddings with the Danish FastText word embeddings from Grave et al. (2018) that were trained on a general domain.

2.4 Datasets

We compared performances between models using four medical datasets: bleeding classification, bleeding site classification, venous thromboembolism (VTE) classification, and VTE site classification. All samples were annotated with a consensus label from three medical doctors. The dataset distributions can be seen in Table 2 and examples of samples can be seen in Appendix C.

2.4.1 Bleeding classification

The bleeding dataset (Pedersen et al., 2021) is a binary classification problem with 25,862 samples. The dataset was constructed from 900 Danish electronic health records (EHR) from Odense University Hospital. The samples had an average token length of 13.3.

2.4.2 Bleeding site classification

The bleeding site dataset (Pedersen et al., 2022b) is a 10-class classification problem with 11,250 unique bleeding-positive samples annotated for the bleeding site. The bleeding site labels were: airways, cerebral, ear-nose-throat, eyes, gastrointestinal, gynecological, internal, skin, urogenital, and unknown. The dataset was constructed from 149,523 Danish EHR notes from Odense University Hospital. The samples had an average token length of 14.4.

2.4.3 VTE classification

The VTE dataset (Pedersen et al., 2022a) is a binary classification problem with 22,528 samples. The dataset was constructed from 94,520 Danish EHR notes from Odense University hospital. The samples had an average token length of 13.8.

2.4.4 VTE site classification

The VTE site dataset (Pedersen et al., 2022a) is a 3-class classification problem with 6,000 VTE-positive samples annotated for the VTE site. The VTE site labels were: airways, lungs, and unknown. The dataset was constructed from 94,520 Danish EHR notes from Odense University Hospital. The samples had an average token length of 14.5.

2.5 Fine-tuning

2.5.1 MeDa-BERT and BERT

We used the [CLS] token followed by a classification layer to classify samples of the datasets. We searched for the best models five times using Adam with learning rates [5e-5, 3e-5, 1e-5], i.e., we fine-tuned each model 15 times. The models were trained for a maximum of 10 epochs.

2.5.2 LSTM

We used the medical word embeddings as input to a bidirectional LSTM layer with a hidden layer size of 512. The last hidden state of the LSTM was followed by a dropout layer with probability 0.2, a dense layer of size 256, a ReLU activation function, a dropout layer of probability 0.2, and a dense classification layer. This model is referred to as LSTM+MeDa-WE.

The performance of the model is compared with another LSTM model (LSTM+General-WE) with the same parameters but using FastText embeddings trained on the general domain as input (Grave et al., 2018). We searched for the best

	Bleeding	Bleeding site	VTE	VTE site
LSTM+General-WE	$83.8_{.7}$	$69.3_{.8}$	$88.5_{.2}$	$86.4_{.7}$
LSTM+MeDa-WE	$\mathbf{91.4_{.3}}$	$\mathbf{84.9_{1.1}}$	$\mathbf{94.1_{.3}}$	$\mathbf{93.4_{.4}}$
BERT	$94.3_{.6}$	$86.7_{.8}$	$96.7_{.3}$	$94.7_{.3}$
MeDa-BERT_16	$94.7_{.3}$	$88.4_{.6}$	$\mathbf{97.1_{.4}}$	$95.5_{.2}$
MeDa-BERT_32	$95.1_{.5}$	$88.7_{.6}$	$96.9_{.3}$	$95.7_{.3}$
MeDa-BERT_48	$\mathbf{95.3_{.4}}$	$\mathbf{89.1_{.2}}$	$97.0_{.5}$	$\mathbf{95.8_{.3}}$

Table 3: Mean accuracy and standard deviation (subscript) for each model on four medical classification tasks. Best results for the LSTM and BERT-based models highlighted in bold. MeDa-BERT_16 denotes the MeDa-BERT model pretrained for 16 epochs.

models five times using Adam with learning rates [5e-5, 3e-5, 1e-5], i.e., we fine-tuned each model 15 times.

For all models we report the mean test set accuracy and standard deviation for the five best performing models on the validation dataset.

3 Results

Table 3 shows the results of each model on the four classification datasets.

3.1 Word embedding comparison

Using the medical word embeddings as input to an LSTM model resulted in large improvements compared to using general word embeddings. On average, LSTM+MeDa-WE outperformed the LSTM+General-WE model by 8.9 percentage points (PP). The largest improvement was seen on the 10-class bleeding site classification with an improvement of 15.6 PP.

3.2 Language model comparison

Comparing BERT and MeDa-BERT, the performance improvements were smaller. MeDa-BERT performed better on three of the datasets with an average improvement of 1.2 PP. The largest improvement was on the 10-class bleeding site classification with an improvement of 2.4 PP.

4 Discussion and limitations

This paper presented a new Danish medical corpus that was used to train NLP models. The corpus included medical books and text scraped from medical websites that provide information for both citizens and healthcare professionals. We applied different techniques to filter out non-medical data, e.g., by removing documents from non-medical departments or text written by non-healthcare professionals. While these steps did remove a large part of non-medical text, some non-medical text might still be present in the corpus. However, the results showed that models pretrained on the medical corpus performed better than general-domain models, especially for multiclass classification problems.

For the Danish language, few medical evaluation datasets are available and therefore the models were only evaluated on classification tasks. Moreover, the evaluation datasets were constructed from EHR text which has its own nuances compared to the text of the medical pretraining corpus, e.g., EHR text contains many spelling mistakes whereas the medical corpus contains few grammatical errors. These factors might limit the generalizability of the results. Future work should evaluate the models on other tasks, e.g., named-entity recognition and question answering which will provide a better understanding of the models' capabilities.

We found continuous small performance improvements by pretraining MeDa-BERT for more epochs. The model might improve with further pretraining but because of limited computational resources and the small rate of improvement, we did not explore this further. The model would also benefit from more medical pretraining data. Although this paper presented a large part of the available medical Danish text, more data could be collected, e.g., from other medical book publishers and websites.

The medical datasets used to evaluate the models are not publicly available because of privacy concerns. For future work, we will strive to publish parts of the medical corpus which requires permission from the text owners. We advise interested researchers to contact us for sharing possibilities.

5 Conclusion

This paper presented a Danish medical corpus consisting of 133M tokens. The corpus was used to pretrain medical word embeddings and language models. The models trained on the medical corpus performed better than similar models trained on a general domain.

Acknowledgement

We would like to thank the medical corpus contributors who gave acceptance of using their text in this project. Listed in alphabetical order: FADL's forlag, Medicin.dk, Netdoktor.dk, Sundhed.dk, The Capital Region of Denmark, The Central Region of Denmark, The Region of Northern Denmark, The Region of Southern Denmark, The Region of Zealand.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. https://doi.org/10.18653/v1/W19-1909 Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. https://doi.org/10.18653/v1/D19-1371 SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. https://aclanthology.org/Q17-1010 Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5.

Édouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomáš Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. https://doi.org/10.18653/v1/2020.acl-main.740 Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. http://arxiv.org/abs/1412.6980 Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Martin Sundahl Laursen, Jannik Skyttegaard Pedersen, Pernille Just Vinholt, Rasmus Søgaard Hansen, and Thiusius Rajeeth Savarimuthu. 2023. Benchmark for evaluation of danish clinical word embeddings. *Northern European Journal of Language Technology*, 9(1).

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jannik Pedersen, Martin Laursen, Pernille Just, Anne Alnor, and Thiusius Savarimuthu. 2022a. Investigating anatomical bias in clinical machine learning algorithms. In *Findings of the European Chapter of the Association for Computational Linguistics: EACL 2023*, Dubrovnik, Croatia. Association for Computational Linguistics.

Jannik S Pedersen, Martin S Laursen, Thiusius Rajeeth Savarimuthu, Rasmus Søgaard Hansen, Anne Bryde Alnor, Kristian Voss Bjerre, Ina Mathilde Kjær, Charlotte Gils, Anne-Sofie Faarvang Thorsen, Eline Sandvig Andersen, et al. 2021. Deep learning detects and visualizes bleeding events in electronic health records. *Research and practice in thrombosis and haemostasis*, 5(4):e12505.

Jannik S Pedersen, Martin S Laursen, Cristina Soguero-Ruiz, Thiusius R Savarimuthu, Rasmus Søgaard Hansen, and Pernille J Vinholt. 2022b. Domain over size: Clinical electra surpasses general bert for bleeding site classification in the free text of electronic health records. In *2022 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 1–4. IEEE.

Bo Peng, Emmanuele Chersoni, Yu-Yin Hsu, and Chu-Ren Huang. 2021. Is domain adaptation worth your investment? comparing bert and finbert on financial tasks. In *Proceedings of the Third Workshop on Economics and Natural Language Processing*, pages 37–44.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. https://doi.org/10.18653/v1/W19-5006 Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. https://doi.org/10.18653/v1/W18-5446 GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing* *and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. https://doi.org/10.18653/v1/2020.emnlp-demos.6 Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Appendices

A Preprocessing of text corpuses

A.1 Medical information portals

Netdoktor.dk provides information about diseases, symptoms, medication, and treatment. Netdoktor.dk contains sections that are not related to the medical domain and discussion forums where users can communicate. Therefore, we removed documents having links containing the following strings: debat, kultur, testdigselv, behandlerguiden, nyhedsbrev, nyheder, privacypolicy, kontaktnetdoktor, cookieinformation, disclaimer, sponsorindhold and discussions. Moreover, citizens can ask medical questions [6] that are answered by health care professionals. We only included the answers to these questions.

Medicin.dk has three sub-pages: `www.min.medicin.dk` that provides information to citizens, `www.pro.medicin.dk` that provides information to health care professionals, and `www.indlaegssedler.dk` that contains information about medicine. We included all documents from these webpages.

Sundhed.dk provides information for medical professionals [7] and citizens [8] about diseases, symptoms, medication and treatment. We included all documents from these webpages.

A.2 Clinical guidelines

We collected clinical guidelines from the 5 regions of Denmark: The Capital Region of Denmark, The Region of Northern Denmark, The Region of Southern Denmark, The Region of Zealand, and The Central Region of Denmark. For each region we removed non-medical documents, seen in Table 4.

Region	Categories removed (in Danish)	Date retrieved	Tokens
Capital Region	Den sociale virksomhed Center for ejendomme Center for HR Center for Regional Udvikling Region Hovedstadens Apotek Steno Diabetes Center Copenhagena	October 2022	13,443,269
Northern Region	Logistik afdeling Teknisk Afdeling Himmerland Teknik Logistik Service	October 2022	6,505,559
Southern Region	Administration Service PsykInfo	September - November 2022	29,075,187
Region Zealand	Administration HR organisation og ledelse Indkøb IT PortørCentral Rengøring Økonomi Uddannelse	November 2022	6,387,083
Central Region		November 2022	25,156,478

Table 4: Categories removed and number of tokens from each of the Danish regions.

Parameter	Value
Architecture	
Number of layers	12
Hidden size	768
FFN inner hidden size	3072
Attention heads	12
Attention head size	64
Dropout	0.1
Attention dropout	0.1
Max seq. length	512
Optimization	
Learning rate	1e-4
Optimizer	AdamW
Adam weight decay	0.01
Adam epsilon	1e-6
Adam beta1	0.90
Adam beta2	0.98
Learning rate decay	Linear
Batch size	4032
Warm up	1 epoch
Epochs	16, 32, 48
Gradient clipping	1.0

Table 5: Architecture and optimization parameters for pretraining MeDa-BERT

B Model parameters and pretrainng loss

Table 5 shows the architecture and optimization parameters for pretraining MeDa-BERT. Table 6 shows the masked language modelling loss for MeDa-BERT during pretraining.

[6] https://www.netdoktor.dk/brevkasser/

[7] https://www.sundhed.dk/sundhedsfaglig/

[8] https://www.sundhed.dk/borger/

	Train loss	Validation loss
MeDa-BERT_16	2.122	2.019
MeDa-BERT_32	1.874	1.792
MeDa-BERT_48	1.766	1.673

Table 6: Masked language modelling loss for MeDa-BERT during pretraining. MeDa-BERT_16 denotes the model pretrained for 16 epochs.

C Dataset examples

Figure 7 shows a sample from each dataset translated from Danish to English.

Dataset	Example	Label
Bleeding	"Girl hospitalized on 14.05.11 with bleeding tendency. 1½ years ago, noticed bleeding on both arms, under the armpits and on the inner thighs. Subsequently blood discharges on the mucous membrane of the cheeks and quite heavy menstrual bleeding, which is unusual for pt."	Positive
Bleeding site	"19-year-old man referred by on-call doctor due to sudden onset of macroscopic hematuria and left-sided flank pain."	Urogenital
VTE	"Pt has severe heart failure and hence dyspnoea and the feeling of air hunger, and, in addition, pt has pulmonary embolisms and COPD. Treatment with Fragmin has started."	Positive
VTE site	"Irregular contours on the left side in the transverse sinus and beginning part of the sigmoid sinus compatible with partial thrombosis."	Brain

Table 7: Example of a sample from each dataset translated from Danish to English.