Universiteit Leiden

Master Computer Science

Investigating De-identification Methodologies in Dutch Medical Texts: A Replication Study of Deduce and Deidentify

Name:	Ruilin Wang
Student ID:	s3096440
Date:	02/06/2023
Specialisation:	Computer Science: Data Science
1st supervisor:	Prof.dr. Marco Spruit
2nd supervisor:	Dr. Pablo Mosteiro Romero (UU)

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

De-identifying sensitive information in electronic health record (EHR) systems is becoming increasingly important as legal obligations to data privacy evolve and the need to protect patient and institutional confidentiality. We conduct research on the task of de-identifying clinical records stored in hospital EHR systems in accordance with Dutch and EU legislations.
This research employs two distinct datasets: the 'Annotation-based Dataset' from the Utrecht University Medical Center, containing a wide array of manually annotated patient records spanning 1987 to 2021, and the 'Synthetic Dataset,' generated using a two-step process involving OpenAI's GPT-4 model. These diverse datasets, although disparate in their origin, real-world and synthetic, provide a comprehensive basis to evaluate de-identification techniques in the context of Dutch medical texts.
The thesis focuses on a comparative analysis of two de-identification techniques, Deduce and Deidentify, considering their performance in de-identifying the Dutch medical texts. Utilizing precision, recall, and F1 scores as evaluation metrics, the research uncovered the relative strengths and limitations of the two methods.
Our findings indicate that both techniques show variable performance across different entities of de-identifying text information. Deduce outperforms Deidentify in overall accuracy by a margin of 0.42 on the synthetic datasets, and on the real-world annotation-based dataset, the generalization ability of Deidentify is lower than Deduce by 0.2. However, the performance of both techniques is affected by the limitations of the dataset.

Keywords: Dutch medical records; privacy information; natural language processing; named entity recognition; machine learning; deep learning methods.

Acknowledgements

As my master's journey draws to a close, I would like to thank the various people who have helped me with my academic and personal development.
My deepest thanks to Prof. Dr. M.R. Spruit for giving me the opportunity to participate in this interesting and challenging project. His quick and insightful responses, provision of relevant resources, coordination with other experts, and unwavering encouragement and guidance were critical to the successful execution of this study.
I am very grateful to Dr. Pablo Mosteiro Romero for his help with the data, coding support, and his generosity in labeling the dataset. He contributed significantly to the quality and depth of my research by sharing his expertise and relevant information.
I would like to express my sincere gratitude to my parents for their unwavering emotional and financial support throughout this journey. They gave me faith and inspired me.
Special thanks to my student coach Drs. A. Blank. Her encouragement and mental support were crucial throughout my academic journey.
This journey has allowed me to grow significantly both academically and personally. It was an experience of self-discovery, adaptation, and continuous learning.
Finally, I humbly acknowledge my limitations and openly welcome constructive criticism for future growth. My heartfelt thanks to everyone involved in this remarkable journey.

Contents

1 Introduction

Electronic health records (EHRs) are invaluable data sources for research in healthcare systems and related fields. They contain a lot of useful information that is important for guiding medical care and advancing medical research. However, due to its sensitivity, the protection of private data is important Meystre et al. (2014).

1.1 Data Privacy in Health Records

To ensure the protection of private data, regulations have been established globally. For instance, in the United States, Protected Health Information (PHI) defined by the U.S. Department of Health Insurance Portability and Accountability Act (HIPAA), protected information is health data that can identify a specific individual. Protecting PHI and broader data privacy in medical records is a critical legal requirement Edemekong et al. (2022).

Similarly, in countries across Europe, the General Data Protection Regulation (GDPR) has set very strict rules to ensure the protection of personally identifiable private information (Voigt and von dem Bussche, 2017). These legislative environments, including the Netherlands, have advanced and accelerated the development of privacy-preserving technologies, such as those dealing with the de-identification of PHI.

1.2 De-identification Techniques

De-identification techniques are essential in the privacy preservation process. Before discussing the different de-identification techniques, we illustrate the process of de-identifying personal information on Dutch-language medical data through a practical demonstration. To this end, we present Figure 1, which illustrates the de-identification of a piece of Dutch medical text. In this figure, the original text contains PHI instances, represented by different categories in the legend (among them are Patient, Persoon, Locatie, Instelling, Datum, Leeftijd, Patientnummer, Telefoonnummer, Url). Then, the text will go through the de-identification process through different NER techniques, finally given that all PHI instances have been removed.

Next, we briefly discuss the main de-identification techniques in our research:

1) **Rule-based Approach:** This method was one of the earliest technologies for removing personal privacy data from electronic medical records and remains widely used today. It uses pre-defined rules, language knowledge, and domain-specific dictionaries to identify or replace PHI in text. Pattern matching, context analysis, and dictionary checklists are often necessary components of this approach. Douglass et al. (2005) used lexical lookup tables, regular expressions, and heuristic methods to de-identify medical care text data in US hospitals. The work of Menger et al. (2018) applied these methods and techniques to a test corpus of Dutch medical care notes and treatment plans. They created and used a pattern-matching method called Deduce to automatically de-identify sensitive information. This approach highlights the usability and customizability of rule-based methods on textual data, especially in the context of the Dutch language in the medical domain.

2) **Artificial Intelligence and Machine Learning:** The emergence of deep learning algorithms has been proven to be applied to natural language processing problems and can also solve the problem of named entity recognition in texts, which means that deep learning techniques and methods can solve the task of removing personal sensitive and private information from medical texts Goodfellow et al. (2016). These methods are trained from labeled data and learn complex textual patterns and contextual structures during training, a process that, unlike rule-based methods, does not require explicit programming. Trienes et al. (2020) conducted a comparative study on the rule-based, feature-based, and deep learning methods for Dutch medical record de-identification. They used the multilingual datasets of different institutions to conduct comparative tests of different methods. Their research direction has accelerated research on the de-identification of medical texts in Dutch, although we found some issues with their evaluation process. Therefore, we believe that their results do not adequately demonstrate the superiority of state-of-the-art neural architectures over rule-based methods in the de-identification of Dutch medical texts.

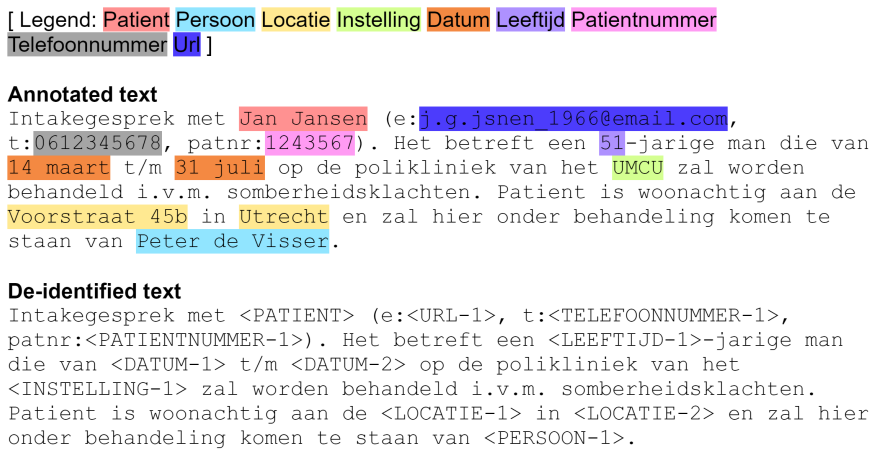

Figure 1: Example of the de-identification process TDSLAb (2023). The figure consists of three parts: the legend categories, the original Dutch medical text containing instances of the legend categories, and the de-identified text with the legend categories removed.

The deep learning algorithm method has been widely used in the de-identification of EHR. For example, the artificial neural network (ANN) is used to remove sensitive information in the patient notes in the US medical database. The evaluation after the application of this technology proves that the neural network can complete the task and does not need the establishment of rule-based methods and feature-based methods Dernoncourt et al. (2016); Ahmed et al. (2020). Undoubtedly, we also found that the evaluation of the results of different methods often depends heavily on the nature of the datasets used for testing. We found that when comparing different de-identification methods, the choice of test dataset can affect the results. This emphasizes the importance of selecting and using appropriate data in EHR research.

1.3 Evaluation of De-identification Methods

Therefore, evaluating de-identification methods is another key factor in this field, and we need to use the same standard data and evaluation methods to process the training results. In the problem of removing PHI medical privacy data in the Named entity recognition problem, the evaluation method of this process usually includes measuring the recall rate (ability to identify all PHI elements) and precision (avoiding non-PHI elements) to provide relevant technologies or models. Recognition performance needs to be objectively assessed. Comparisons between different models should be made on the same dataset Aberdeen et al. (2010).

1.4 Aims of this Paper

This paper is fundamentally a replication-extension study aimed at comparing de-identification methodologies in Dutch medical data. Building upon and extending the research carried out by Trienes et al. (2020) that investigated rule-based Menger et al. (2018) and neural network approaches, our main research question is:

> *'How do the rule-based and neural network approaches to de-identification compare in terms of effectiveness, adaptability, and generalization when applied to Dutch medical data, and how can the strengths of these methodologies be used to improve the de-identification process and enhance patient privacy protection?'*

We plan to address this question by conducting comparative experiments using correctly labeled datasets and exploring the feasibility of both rule-based methods and deep learning algorithms for de-identification in Dutch medical texts. Our aim is to provide clear, accurate results and insights into effective de-identification techniques by employing an effective assessment methodology. Ultimately, this research aims to contribute to the body of knowledge enriching privacy protection in the medical records of the Netherlands. By exploring established and emerging de-identification methods, we hope to offer a comprehensive view of the field as it stands today and provide insights for future improvement and development.

This thesis is structured as follows:

- In Section 2 Related Work, we review related work in the different domains of rule-based methods, feature-based methods, deep neural network methods, and the comparisons between these methods in some domains.
- In Section 3 Experimental Design, we discuss our experimental design, starting with a description of the dataset, followed by rule-based and neural network methodology, which includes the approach of Deduce, Deidentify(bilstmcrf). We also define the evaluation metrics that we will use.
- In Section 4 Results, we present our results, including model performance and a comparative analysis of the different methods of our experiment.
- Finally, in Section 5 Conclusions, we draw conclusions from our work, providing a summary and outlining directions for future work.

2 Related Work

In this section, we cover the various approaches used for the De-identification of medical records which include rule-based, feature-based, deep learning method, and their evaluation methods. In order to systematically explore and analyze the vast body of existing research on these methods, we employed the SYMBALS literature review methodology. SYMBALS is a systematic review methodology that integrates traditional backward snowballing with active learning. This approach expedites the process of title and abstract screening, providing efficient coverage in rapidly expanding research fields van Haastrecht et al. (2021). This methodology is implemented in ASReview, more on ASReview: https://asreview.nl/.

2.1 Data Privacy and De-identification in Health Records

In this subsection, we review the literature related to data privacy and de-identification in health records. These literature reviews provide a comprehensive overview of the current state of research in this field and highlight the importance of de-identification in protecting data privacy in the healthcare industry.

The long-recognized necessity of maintaining privacy for personal data in health records does not preclude the ability to extract useful information from these records. This issue has received considerable attention recently due to the increasing volume and importance of electronic health records (EHRs) and other digital health data. An example is the work of Tikk and Solt (2010), which focuses on extracting de-identified drug information from health records, demonstrating the practical application of de-identification methods in a healthcare setting.

One way to achieve the task of de-identification is to use natural language processing (NLP) techniques to automatically identify and classify personal health information (PHI) categories, which is an important step to protect data privacy. The work of Yang and Garibaldi (2015), who designed an NLP processing system for the 2014 i2b2 de-identification Challenge. Their task focused on identifying and classifying seven major PHI categories and 25 related subcategories, and they emphasized the complexity and diversity of data that needed to be addressed in the

de-identification process.

Furthermore, de-identification is not simply to extract or delete PHI. It can also protect data privacy by identifying and categorizing clinical terms more accurately. As Wang et al. (2019) clarified, clinical named entity recognition involves the identification and classification of clinical terms such as disease, symptom, treatment, examination, and body part in EHR. Accurate identification and categorization of such terms play a key role in ensuring data privacy and de-identification, as it helps to distinguish between patient-identifiable information and general medical terms, which indirectly contributes to the de-identification process.

These studies demonstrate multiple aspects and techniques for de-identification and data privacy in health records. However, the success and accuracy of these techniques vary depending on the specific methodology used, the quality and complexity of the data, and the specific context in which they are applied. This illustrates the importance of ongoing research and evaluation in this area, which we explore in more detail in the following sections.

2.2 Techniques for De-identification

The field of de-identification has proposed a range of approaches, each aimed at protecting individual patient privacy while maintaining the usefulness of health records in research and clinical settings. These techniques have advanced considerably over time, and more and more attention has been focused on developing hybrid models. Some models use rule-based systems and machine learning-based methods to effectively de-identify sensitive data Yang and Garibaldi (2015).

The scope of de-identification techniques is generally divided into three main methodologies: rule-based, feature-based, and deep neural networks. Each method has its unique strengths and challenges, and the choice of method is often task- and situation-specific.

2.2.1 Rule-Based Method

De-identification methods based on rule-based methods are one of the early applications in this field. Its main strength lies in establishing explicit rules to encapsulate the understanding of the problem, thus enabling human readability and interpretability. Furthermore, this method does not require a slow training phase, making it very beneficial in time-sensitive scenarios Tikk and Solt (2010).

At the core of rule-based de-identification of textual content are regular expressions, one of the template patterns designed to characterize the semantics of various categories of personal health information (PHI). These patterns facilitate the identification of sensitive data points and their subsequent modification or masking, ensuring the safety of patient privacy Yang and

Garibaldi (2015).

In the context of the de-identification of Veterans Health Administration (VHA) clinical documents, Ferrández et al. (2012) evaluated a rule-based approach against a machine learning approach. In their study, the rule-based system demonstrated better recall, indicating a greater ability to identify all PHI instances correctly.

However, the accuracy (representing the proportion of correctly identified instances out of all identified instances) of rule-based approaches was found to be lower compared to machine learning techniques. Therefore, while rule-based de-identification can successfully identify most PHI, it can also incorrectly identify non-PHI as sensitive information. Therefore, adopting a rule-based approach must carefully balance the trade-off between recall and precision depending on the use-case requirements and the acceptable margin of error Ferrández et al. (2012).

Although there are challenges, rule-based methods are a viable option for de-identification due to their understandable nature and comparable performance to more complex machine learning methods. This is especially true for large datasets used in machine learning training, where labeling language and domain may not be feasible Tikk and Solt (2010). Future research could focus on enhancing the accuracy of these methods and integrating them into hybrid de-identification systems to improve overall performance.

2.2.2 Feature-Based Method

The research on feature-based de-identification methods is promising, as it utilizes various linguistic properties to identify and eliminate personally identifiable information. These techniques rely heavily on syntactic and word-surface-oriented features, and are further improved with task-specific features Yang and Garibaldi (2015).

One well-known feature-based system is the Medical Language Extraction and Encoding System (MedLEE), which has been utilized in numerous studies. MedLEE makes use of various linguistic and semantic characteristics to detect sensitive information in clinical documents. While it is effective, its performance relies heavily on the quality and relevance of the input features, which may present difficulties in diverse or complicated datasets Morrison et al. (2009).

Tikk and Solt (2010) demonstrated the potential of feature-based methods by comparing conditional random field (CRF) models (a type of feature-based approach) to rule-based methods. Initially, the CRF model did not perform better than rule-based methods due to insufficient training data. However, after adding more training data (even if not entirely accurate), the performance of the CRF model improved significantly. This highlights the significance of a diverse and comprehensive training dataset in enhancing the performance of feature-based

models.

Exploring different models, even outside the field of health informatics, can provide valuable insights for improving de-identification techniques. A prime example is the Retrieval Enhanced Transformer (RETRO), a language model that has shown impressive performance in data- and knowledge-intensive tasks at a large scale. RETRO is a unique model because it uses its built-in memory and a large collection of text, known as 'corpora', to search for and get useful parts from this vast collection of text, a process known as 'document chunk retrieval'. This ability allows RETRO to present a new method for de-identification that specifically targets and removes features that could identify someone. While its effectiveness in de-identifying health records remains to be tested, its potential for processing vast amounts of health data opens up possibilities for further research in this field Borgeaud et al. (2021).

Collectively, these studies show that feature-based de-identification methods have immense potential. These methods are capable of identifying and removing sensitive information by extracting and utilizing intricate linguistic patterns. However, their effectiveness depends on the quality and scope of the features and the diversity and size of the training dataset. To improve feature-based de-identification, future research can explore new feature sets, expand and diversify the training data, and incorporate insights from related domains.

2.2.3 Deep Neural Network Method

Deep neural networks (DNNs) are increasingly at the forefront of de-identification due to their ability to model complex patterns in large datasets. They have been widely used in natural language processing tasks, including those related to de-identification Stubbs et al. (2015).

One pioneering work in this field is the de-identification system based on artificial neural networks (ANN) introduced by Dernoncourt et al. (2016). What sets this system apart is that it does not need feature engineering, but still outperforms earlier systems, achieving an impressive F1 score of 97.85 on the i2b2 2014 dataset and 99.23 on the MIMIC de-identification dataset.

On this basis, Khin et al. (2018) developed a dedicated deep learning architecture for the de-identification of patient notes, which reinforces the effectiveness of deep neural networks in this area.

Recent research has begun to explore the integration of deep learning with other computational methods to improve predictive accuracy on complex datasets. For example, Islam et al. (2020) demonstrate an innovative associative memory-based deep learning method integrated into a belief rule-based expert system (BRBES) called BRB-DL. It is found that adding deep processing layers, including associative memory, to BRBES can improve prediction accuracy,

outperforming traditional deep learning methods.

When it comes to language processing tools, Stanza, an open-source Python NLP toolkit, is highly regarded for its fully language-agnostic neural pipeline. It can handle important tasks like tokenization, lemmatization, part-of-speech tagging, and named entity recognition, which are crucial in the de-identification of health records Qi et al. (2020).

In addition to these, Agnikula Kshatriya et al. (2021) proposed a novel approach to identifying asthma control factors in clinical notes, particularly inhaler techniques, using a hybrid deep learning model. Their model employs the original BERT and a specialized clinical BioBERT (cBERT), along with post-hoc rules to reduce model errors. Additionally, their approach introduces distant supervision. This technique leverages weak labels, which are heuristic-based or lower-quality information, to supplement the training data. By employing weak labels, they manage to increase the size of the dataset used for training their model without the need for laborious manual labeling, hence eliminating the associated high costs. The strategy of using weak labels, in this context, provides a cost-effective solution for data augmentation, allowing their deep learning models to learn from a more extensive dataset. Consequently, the BERT models trained with distant supervision outperformed both the rule-based model and the BERT models trained on original data, demonstrating the effectiveness of combining deep learning models with other methodologies for specific tasks in de-identification.

Moreover, new technologies such as BERT and multi-layer perceptron (MLP) can be applied in this field. BERT embeddings have been used for medical linguistic reasoning, and MLPs for textual entailment recognition. Strategies such as 5-fold stacking have been used to learn and combine predictions, helping to develop more accurate de-identification systems Tawfik and Spruit (2019).

Deep learning has shown potential in the specific task of clinical named entity recognition, which is the basis of clinical and translational research. For example, the model proposed by Wang et al. (2019) combines a data-driven deep learning approach and a knowledge-driven dictionary approach to handle rare or unseen entities in electronic health records efficiently. The model extends the bidirectional long short-term memory neural network and employs five different feature representation schemes, demonstrating competitive performance.

In conclusion, while deep neural network approaches show great promise in the field of de-identification, it is important to consider remaining challenges, including the need for large labeled datasets, computational requirements, and difficulties in interpretability. However, the applications of these methods are evolving, and with further research, they may provide powerful tools to aid in the task of de-identifying health records.

This part reviews and details various techniques used in medical record de-identification, including rule-based, feature-based, and deep learning approaches. The Table 1 summarizes each method and its technique as follows:

Table 1: Summary of the Technical Methods Included in Each Approach

Technical Methods	Rule-Based	Feature-Based	DNNs
Regular Expressions Yang and Garibaldi (2015)	✓		
Dictionary Lookup Menger et al. (2018)	✓		
Custom Rules Ferrández et al. (2012)	✓		
ML Algorithms Morrison et al. (2009)		✓	
CRF Tikk and Solt (2010)		✓	
Bidirectional LSTMs Wang et al. (2019)			✓
Transformer Agnikula Kshatriya et al. (2021)			✓
BiLSTM-CRF Trienes et al. (2020)			✓

- **Rule-Based Method:** These traditional approaches center on the construction of precise rules aimed at detecting and anonymizing personally identifiable information (PII) in medical data. Despite their fundamental role and great potential, rule-based approaches face many challenges, including dealing with variations and anomalies that often occur in real-world datasets.
- **Feature-Based Method:** Such de-identification techniques utilize machine learning algorithms and natural language processing (NLP) mechanisms. Feature-based methods have attracted great interest due to their flexible nature and ability to manage large amounts of data. Their pattern recognition and machine learning aspects show remarkable promise, especially when configured for specific document structures and PII types. However, current research suggests opportunities for further refinement and optimization, emphasizing the need for customization and adaptation to improve its effectiveness.
- **Deep Neural Network Method:** The application of deep neural networks (DNN) in the field of de-identification has shown great potential. DNN approaches have demonstrated strong capabilities to identify and anonymize PII, often outperforming traditional rule-based and feature-based approaches. They also demonstrate versatility across different languages and domains. Nonetheless, challenges remain, especially in terms of diverse datasets and domain-specific features. These challenges underscore the ongoing need for continuous research and improvement of DNN methods.

2.3 Evaluation and Comparison of Deidentification Methods

In order to effectively evaluate and compare various de-identification methods, it is essential to have a good understanding of key evaluation metrics and processes. Metrics such as precision and recall are cornerstone metrics for evaluating the effectiveness of de-identification techniques. Ensuring that these measurements are derived from a consistent dataset is a critical

step in maintaining reliable and comparable estimates Khin et al. (2018).

An in-depth analysis of multiple de-identification systems shows that systems employing rule templates as features and incorporating statistical learning methods usually yield the best results. Hybrid systems follow in terms of effectiveness, followed by purely machine learning-based and rule-based systems Uzuner et al. (2007). However, challenges remain in enhancing these systems to maintain the validity of various data types and managing policy-driven concerns about automated de-identification methods.

The performance of various methods can differ significantly depending on the task at hand. For instance, a medical language reasoning task achieved an accuracy score of 0.85 by using BERT context embeddings and machine learning techniques. On the other hand, a classical multilayer perceptron network for textual entailment recognition has a relatively low accuracy score of 0.58, as reported by Tawfik and Spruit (2019).

In a study conducted by Tikk and Solt (2010), they compared rule-based methods to conditional random field (CRF) models and found that the performance results were similar. It is worth noting that the performance of the CRF model was enhanced with the use of additional training data.

The evaluation methods used in one study are based on various Natural Language Processing (NLP) tasks and datasets, including question answering (SQuAD 1.1 and 2.0), coreference resolution (OntoNotes), relation extraction (TACRED), and a general language understanding evaluation benchmark (GLUE) Joshi et al. (2019). The SpanBERT model displayed results, achieving high F1 scores on the SQuAD tasks and setting a new state-of-the-art on the OntoNotes coreference resolution task. The study revealed performance gains across a variety of tasks, particularly those that involve span selection.

A recent study has introduced two new variants of coreference resolution evaluation metrics, B3sys and CEAFsys Cai and Strube (2010). These metrics aim to improve the accuracy of evaluating end-to-end coreference resolution systems. Unlike previous metrics, B3sys and CEAFsys can handle system mentions, which are a common part of these systems. The study's results demonstrate that these new metrics provide intuitive and reliable results, making them essential for comparing different end-to-end coreference resolution systems accurately.

Yang and Garibaldi (2015) conducted an evaluation study that demonstrated an exceptional method achieving a high micro-averaged F-measure of 93.6% in the 2014 i2b2 de-identification challenge.

Research by Wang et al. (2019) introduces a model that integrates a dictionary into a deep neural network. The model shows better precision and recall metrics than other approaches for clinical named entity recognition in electronic health records. Their findings suggest that their approach has the potential to enhance the effectiveness of de-identification methods in

medical data privacy.

Although comparative studies offer valuable insights into the effectiveness of different methods, choosing an appropriate method for each task carefully is crucial, considering the application's specific requirements and limitations, regular evaluation and comparison of methods are necessary to advance the field of de-identification research. We present Table 2 below to highlight each method's type and key metrics.

Table 2: Summary of Evaluation and Comparison of Deidentification Methods

Method/Model	Type	Key Metrics
BERT context embeddings	ML-based	Accuracy
Multilayer Perceptron Network	ML-based	Accuracy
SpanBERT	ML-based	F1 scores
Rule Templates	Hybrid	Precision, Recall
2014 i2b2 de-identification method	Hybrid	Micro-averaged F-measure
Model integrating dictionary into DNN	Hybrid	Precision, Recall

2.4 Issues in Current Research and Aims of this Paper

In a recent study, a comparison between Deduce Menger et al. (2018) and Deidentify Trienes et al. (2020) took center stage in Dutch de-identification. Deduce, a rule-based method primarily developed by Menger has been the main anonymization tool at UMC Utrecht. Access to the Deduce code via Github opens the door to transparency and continued development in the field. However, the inherent limitations of rule-based approaches pose challenges. The rigidity of these methods and their reliance on specific data formats affect their adaptability, especially in the face of datasets that deviate from familiar structures. This is further exacerbated by Deduce's heavy reliance on blacklists and whitelists, which include items such as common names and names of Dutch clinical institutions.

To deal with these limitations, Nedap developed another packaged tool for deidentification, Deidentify. Its main distinguishing feature is its adaptability, as it includes models trained on data from specific clinical institutions. Trines et al. show that the performance of the latter is significantly improved. The superior performance of Deidentify is mainly attributed to the rule-based constraints observed in Deduce.

In reviewing the comparative study conducted by Trienes et al., it becomes apparent that the research is undermined by two significant shortcomings. Firstly, the study only examined the generalizability of Deduce, leaving Deidentify's general applicability unchecked. This unbalanced evaluation potentially creates a perception overemphasizing the adaptability of Deidentify. The issue lies in the fact that by not testing Deidentify's ability to function across

different contexts, we may falsely conclude that it is superior, even when Deduce might perform equally well or better in situations not covered by the study. Therefore, the lack of a comprehensive generalizability check for both systems could lead to an inflated appraisal of Deidentify's performance, inadvertently undervaluing Deduce's potential. Secondly, the annotation strategy that Trienes et al. adopted differs from what is typically used in Deduce. This is a significant issue, as annotation plays a crucial role in different models, enabling them to interpret and learn from data effectively. Applying an annotation strategy that Deduce is not optimized for could compromise its ability to demonstrate its full potential, consequently reducing its comparative performance. In essence, the study's altered methodology can create an unfair playing field, tilting the results in favor of Deidentify. These two critical flaws, the different generalizability check and the altered annotation strategy, could cause the original study to favor Deidentify unintentionally. Therefore, a new study is warranted to ensure a more balanced and fair evaluation of the two systems, truly reflecting their respective effectiveness, adaptability, and generalization.

This paper aims to address these limitations in the current research and provide a more objective comparison of Deduce and Deidentify. By examining these systems under controlled conditions, we hope to make an unbiased assessment of their capabilities. The goal is not only to determine which system performs better but to glean insights from both, which can guide the development of new, more efficient anonymity systems. By balancing the strengths of Deduce and Deidentify, this research aims to address the current problem of de-identification of Dutch medical data and bring the field closer to a robust and adaptable solution that preserves patient privacy.

3 Experimental Design

In this section, we delve into the details of our experimental design. This comprises a comprehensive description of the dataset used, followed by the methodology we applied, which includes rule-based and neural network approaches. Furthermore, we also define the evaluation metrics used to measure the performance of our models.

3.1 Dataset Description

This experiment relies on two dataset sources, each generated by a different method, to study de-identification methods in Dutch medical texts.

The initial dataset format used for this experiment is a collection of .jsonl (JSON Lines text format, from UMC Utrecht EHRs system) files. This format is a convenient choice for working with large datasets since each row is a complete and independent data object.

The data consists of Dutch text, metadata, and span tags, reflecting various key information about individuals. Each row in the dataset corresponds to a unique record, formatted as a JSON object. The main fields in the JSON object are "text", "meta" and "spans".

The "text" field contains a string of text in Dutch, usually representing a medical context. It includes patient information such as name, email address, phone number, patient number, facility location, appointment date, and name of the treating physician.

The "meta" field contains additional metadata about the record, such as the individual's first name, last name, initials, data source, year, and unique identifier (uid).

The "spans" field is an array of objects, each object containing information about a particular tag present in the text. Each object consists of the start and end indices of the tags in the text, and the tags themselves. Tags represent various types of identifiable information, such as "patient", "person", "location", "institution", "date", "age", "patient number", "phone number", and "url".

In this study, two de-identification methods, Deduce and Deidentify, are being compared.

Deduce is a rule-based approach created to detect and label entities of a specific format. It is worth noting that its anonymization strategy is unique and clear. For example, it anonymizes the date and the appended year. It does not anonymize the independent year, it will identify them together: 17-01-2023. This design choice was based on discussions with experts in their field. Therefore, for any other system to prove superior to Deduce, it would either need to implement a more efficient anonymization strategy, or use Deduce's strategy and reduce errors.

In contrast, the Deidentify paper proposes a different anonymization strategy, but does not convincingly demonstrate its superiority over Deduce methods. It then compares the performance of Deduce with its gold-standard annotations following different anonymization strategies. Therefore, this study may be seen as unfairly judging Deduce's performance. For example, if Deidentify's data annotation strategy only included the anonymization of individual years (for example only label 2023), the position of predict label would be different from the true label, and it might classify Deduce's approach as flawed for not doing the same. However, this is an intentional design choice, not a bug in Deduce.

Therefore, for a fair comparison between Deduce and Deidentify, it is crucial to label the data using the Deduce labeling strategy. This consideration will significantly affect the selection and design of the datasets used for this experiment.

3.1.1 Annotation-based Dataset

The first source of this dataset was prepared in collaboration with Dr. Pablo Mosteiro Romero, using an annotation strategy inspired by the design of Deduce, a rule-based de-identification tool. The labeling strategy is designed to complement Deduce. This strategy is detailed in Menger et al. (2018), involving multiple rounds of document annotation by students and Dr. Mosteiro.

Prodigy software licensed from UMC Utrecht is used for document annotation. Specifically, Prodigy version 1.10.8 was employed with the recipe "ner.manual" in token mode. This means that each annotation consists of one or more tokens, and partial-token annotations are disallowed. The annotation categories adopted align with those specified by Deduce, as referenced in Menger et al. (2018). The software facilitates data annotation and is particularly useful for refining annotation strategies by identifying inconsistencies. The strategy was honed over multiple rounds in which students and Dr. Mosteiro annotated 10-30 documents, followed by a team discussion of inconsistencies.

The volume of annotated data is directly proportional to the time students dedicate to the annotation process. As a guideline, Trienes et al. (2020) used around 35k sentences, and they found an improvement over Deduce with even 10% of that number. To ensure uniformity in annotation, Dr. Mosteiro also reviewed and annotated a subset of these documents.

A particular challenge addressed in the process is whether to annotate complete or partial

words. Prodigy's design makes whole-word tagging easier (that words do not have clear spaces between them, for example, the sentence 'have a nice day' in Chinese is '祝你今天过得愉快', it does not have clear spaces between words), but partial-word tagging is more in line with Deduce's design (that words have clear spaces between them in a sentence).

3.1.2 Synthetic Dataset

The second source of the dataset was synthetically generated using OpenAI's GPT-4 model OpenAI (2023) and manual annotations. This two-step approach aims to create comprehensive patient medical data in Dutch in a consistent and specific format.

In the first step, a GPT-4 model is used to generate comprehensive patient medical data in English. The data generated consists of various key elements such as patient's name, email address, phone number, patient number, specific locations in the Netherlands, real Dutch institutions, appointment dates, and age.

For this process, the prompt given to the GPT-4 model is: "I want you to help me to generate medical text. Within this medical text, it should include eight categories. Here is the example data [provide the example of correct data]:

```
Intake interview with Jan Jansen (e:j.g.jsnen_1966@email.com, t:
0612345678, patnr: 1243567). It concerns a 51-year-old man who will be
treated at the outpatient clinic of the UMCU from 14 March to 31 July
due to a heart attack. gloom complaints. The patient lives at Voorstraat
45b in Utrecht and will be treated here by Peter de Visser.
```

Please generate 10 sets of such data for me." This procedure is then repeated until a total of 100 sentences have been produced.

After generating the English dataset, in the second step, the text is translated into Dutch by Google Translate. The translated data is then fed to the GPT-4 model to generate a corresponding JSON file. These JSON objects contain patient information in the "text" field, additional metadata in the "meta" field, and indices marking the start and end of each tag within the "text" string in the "spans" field.

To illustrate the structure and characteristics of the synthetic dataset, we give an example:

```
{
 "text": "Jane Doe, een vrouw van 45 jaar oud, met het patiëntnummer 51342
 67 en bereikbaar via j.doe_1978@eposta.nl en 0617356428, zal van 17 mei
 tot 30 juni zorg ontvangen bij het UMC Groningen, Hanzeplein 1, 9713 GZ
 Groningen.",
 "meta":
  {
   "VOORLETTER": "J",
```

```
   "VOORNAAM": "Jane",
   "ACHTERNAAM": "Doe",
   "bron": "rapportages",
   "jaar": 2023,
   "uid": 0
  },
 "spans":
  [
   {"start": 0, "end": 8, "label": "PERSOON"},
   {"start": 184, "end": 215, "label": "LOCATIE"},
   {"start": 169, "end": 182, "label": "INSTELLING"},
   {"start": 127, "end": 145, "label": "DATUM"},
   {"start": 24, "end": 26, "label": "LEEFTIJD"},
   {"start": 57, "end": 64, "label": "PATIENTNUMMER"},
   {"start": 107, "end": 117, "label": "TELEFOONNUMMER"},
   {"start": 83, "end": 103, "label": "URL"}
  ]
}
```


In these examples, each JSON object represents a synthetic patient record. The "text" field contains sentences that might appear in a health record containing a patient's personal and medical details. The "meta" field includes metadata about the patient's name and the source of the information. The "span" field lists the start and end index and corresponding label for each identified entity (patient name, location, etc.).

Labels used in the "spans" field include "PATIENT", "PERSOON", "LOCATIE", "INSTELLING", "DATUM", "LEEFTIJD", "PATIENTNUMMER", "TELEFOONNUMMER", and "URL". Although the GPT-4 model is powerful, we found that the model fails to correctly mark the start and end indexes of each record.

Therefore, a manual annotation process was performed. Wrote and executed a custom Python script to accurately identify the start and end index of each label. The script completes the generation process by correctly annotating the translated Dutch medical text.

As a result, a synthetic Dutch healthcare dataset[1] was successfully created, containing information on 100 patients. The data is synthetic and not real, which helps avoid privacy and confidentiality issues associated with real patient data. However, it should be noted that this data should not be used for real-life medical decision-making as it is entirely generated by machine learning models and has no relation to real-world medical conditions or scenarios.

This method provides a useful method for generating comprehensive medical datasets in Dutch or any other language supported by the model. It protects the privacy of real individuals

[1]Synthetic Dataset: this dataset has its own limitations, it is like how doctors scribble notes, but in standard. No many acronyms and spelling errors, etc.

and supports the development and testing of natural language processing (NLP) tools for healthcare. However, it should be remembered that GPT-4 does not understand text the way humans do. As such, it should not be relied upon solely to accurately label data or make medical decisions.

3.2 Methodology

3.2.1 Deduce

DEDUCE Menger et al. (2018) is an automated method for the de-identification of Dutch medical texts. In this context, de-identification refers to the removal or annotation of any information that can be used to identify an individual in order to protect people's data privacy and comply with local laws and regulations. This is important in a healthcare environment where privacy is paramount and Europe (GDPR) has strict regulations on the processing of personal data.

DEDUCE selects the Protected Health Information (PHI) category: The first step in this process is to determine which types of information in the text data need to be de-identified. This is done in collaboration with medical staff who are very knowledgeable about what information can be used to identify a patient. PHI includes name, address, social security number, etc., as follows:

1. Person names, including initials
2. Geographical locations smaller than a country
3. Names of institutions that are related to patient treatment
4. Dates
5. Ages
6. Patient numbers
7. Telephone numbers
8. E-mail addresses and URLs

De-identification method: Once the type of PHI to be protected is identified, DEDUCE uses a combination of lookup tables, decision rules, and fuzzy string matching to de-identify sensitive information in text data. For example, a lookup table can replace names with pseudonyms. Decision rules may involve rules about which information to de-identify based on context in the text. Fuzzy string matching allows the system to identify and de-identify PHI even if it

is not exactly the same as the information in the lookup table (for example, if there is a misspelling or the spelling is different).

DEDUCE is an automated method for preserving individual privacy in medical texts, which is especially important for research purposes. It does this by identifying and de-identifying PHI, and has been shown to be very effective in test environments. Figure 2 shows an overview of the de-identification method DEDUCE. Next, we introduce the annotation process of each category inside DEDUCE:

- **Person Names:** The method of de-identifying personal names first preprocesses the text to normalize it. In the non-context-based approach, fuzzy string matching matches names to tokens, and shorter names require an exact match. Patient names are annotated according to three rules regarding first name, last name, and initials. A lookup list of common Dutch names, prepositions, and prefixes is used to annotate other names, and a whitelist is used to prevent over-annotation. Context-based methods recognize names based on prefixes and prepositions and use additional rules to annotate adjacent tokens. The method can effectively distinguish patient names from unknown names and de-identify a large portion of the name corpus. For instance, 'Jan Jansen' is annotated as <PERSOON>.

- **Geographical Locations:** DEDUCE annotated all addresses in the data for deidentification, including street names, house numbers, zip codes, and places of residence. Create a place of residence listing using Dutch cities, villages and major European cities. Street names are annotated with a regular expression matching Dutch suffixes. At this point, trie-based hashing techniques are used to handle positions. In this process, sequences of tokens, which might represent a place, a street, etc., are stored in a trie, a type of search tree that effectively manages common prefixes. These sequences are also hashed and stored in a hash table. When a new address needs to be deidentified, it's broken down into tokens, and the trie is used to find matching sequences quickly. The hash of the full sequence is then used to retrieve the anonymized/deidentified version of the address. This efficient method allows for quick and accurate matching of data. Postal codes are matched using regular expressions covering a variety of formats, while house numbers are matched against regular expressions when preceded by annotated street names. Attaches notes to mailbox numbers in a specific format. For instance, 'Voorstraat 45b, Utrecht' is annotated as <LOCATIE>.

- **Names of Institutions:** The DEDUCE method identifies and annotates the name of the care facility where the patient may have been treated. Sources for these designations include internal data, local psychiatric care facilities, large national institutions, and institutions identified during method development. This approach is dataset-specific and users can create suitable lists for their own data. To account for changes in institutional nomenclature, the list was modified to include names without prepositions or articles, abbreviations for multiword institutions, and common word substitutions. The final list contains 742 values and is processed using a trie-based hashing method for computa-

tional efficiency. Any institution names found are annotated, for example, 'UMCU' is annotated as <INSTELLING>.

- **Dates and Ages:** DEDUCE solves the de-identification problem of dates and ages. Dates can be identified, such as dates of birth or hospital admission, and combinations of days and months are dangerous. Using a specific pattern, for instance, '14 maart' is annotated as <DATUM>. While HIPAA guidelines recommend that only ages over 85 need to be de-identified, this approach removes all ages to ensure privacy further, as these ages could be combined with other data to reveal a patient's identity. Use a simple regular expression pattern to detect age, for example, inside '51-jarige', '51' is annotated as <LEEFTIJD>.

- **Patient Numbers:** DEDUCE solves the problem of de-identification of patient numbers. Although they cannot directly identify patients, they may allow connection to other data sources using patient numbers. Since the only identifiable structure in patient numbers is that they are usually seven digits, all seven digits are marked as <PATIENTNUMMER>.

- **Telephone Numbers, Email Addresses and URLs:** DEDUCE tags phone numbers, email addresses, and URLs because of their potential to identify patients directly. Various regular expressions are fine-tuned for each type to ensure efficient annotation. Telephone numbers with ten digits are marked as <TELEFOONNUMMER>, while email addresses and URLs are annotated as <URL>[2].

[2]Note: email address and URL are annotated as 'URL' is the rules of Deduce.

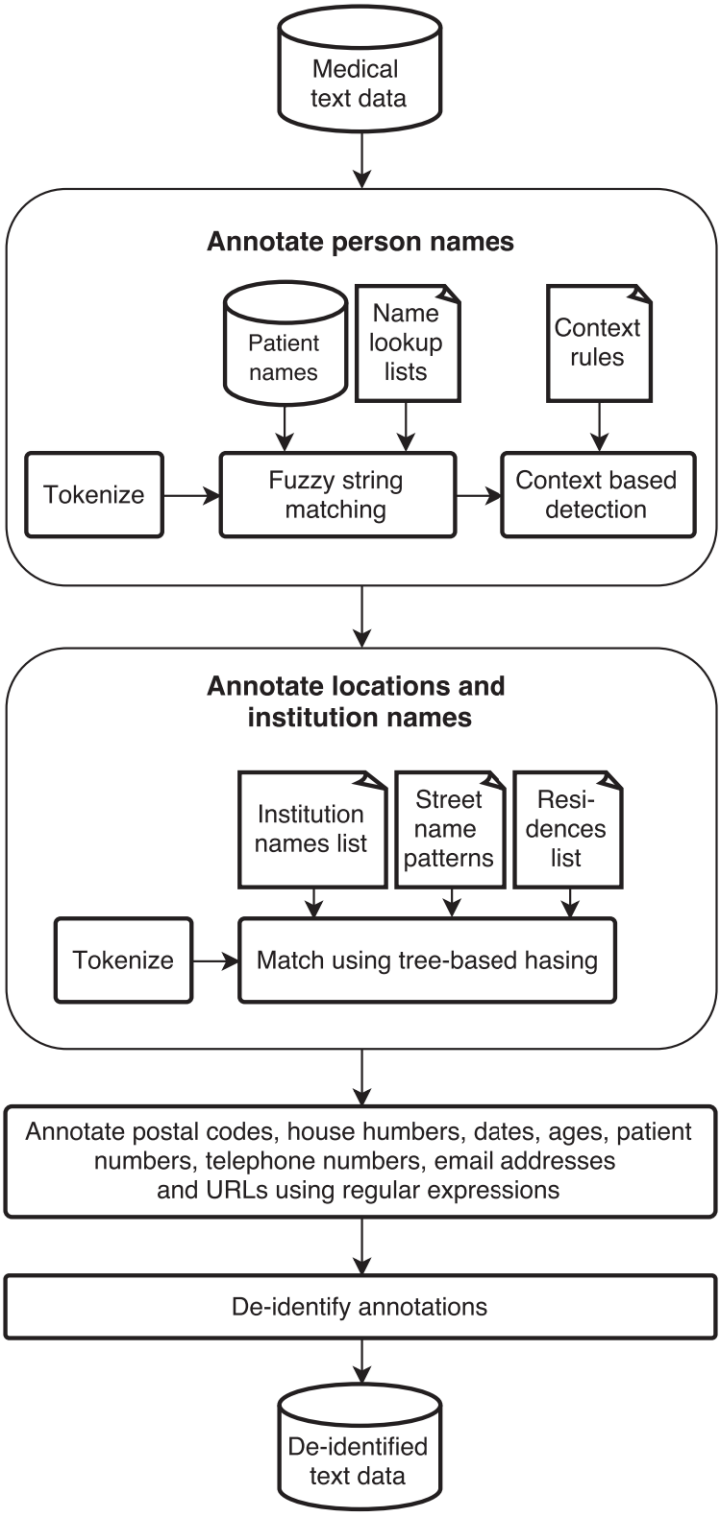

Figure 2: An overview of the de-identification method DEDUCE Menger et al. (2018).

3.2.2 Deidentify

Deidentify Trienes et al. (2020) represents a valuable experimental approach in the field of automatic de-identification of health records, offering a practical solution to the challenge of preserving patient confidentiality while exploiting the wealth of information in unstructured data. Despite the urgent need for such techniques, most research to date has been largely limited to English-language medical texts, leaving gaps in the understanding of the generalizability of de-identification methods across languages and domains. In response to this, the research included in the paper "Deidentify" advances by testing the transferability of derecognition methods across different languages and domains. In particular, we use the pre-trained model of Deidentify 'model_bilstmcrf_ons_fast-v0.2.0' to handle this complex task efficiently.

The Deidentify methodology stands out by leveraging a neural approach termed BiLSTM-CRF (Bidirectional Long Short-Term Memory - Conditional Random Field). The primary advantage of employing this approach is that it minimizes the reliance on hand-crafted features intrinsic to traditional CRF-based de-identification. As detailed in the paper, the BiLSTM-CRF architecture is paired with contextual string embeddings, which have been shown to deliver state-of-the-art results for sequence labeling tasks.

To provide a clearer understanding of the BiLSTM-CRF model, we outline its key steps below:

1. **Convert Standoff Input to Flair Sentences:** The script uses the spaCy tokenizer to convert input data (into Standoff format[3]) into Flair sentences. It is recommended to use the same version of spaCy that was used to train the de-identification model to ensure good compatibility with pre-trained models. The Flair library uses Flair sentences for training and prediction.

2. **Embeddings:** Before passing the data into the model, the script converts the words in the sentences into embeddings. Embeddings are numerical representations of words that capture their semantics.

3. **Initialize/Load the BiLSTM-CRF Model:** The script either initializes a new BiLSTM-CRF model for Named Entity Recognition (NER) or loads an existing one. This model is defined with a hidden size of 256, which designates the capacity of the internal layers to learn and store information during processing. The BiLSTM-CRF model combines a bidirectional LSTM (for capturing sequential information) with a conditional random field (for making sequence predictions, such as tagging entities).

4. **Training/Fine-tuning the Model:** This is the actual learning process where the model is trained to perform NER. During this process, the model is trained for a maximum of 150 epochs, adjusting the model weights based on the training data to minimize prediction errors. Each epoch represents a complete pass through the entire training dataset. Fine-tuning involves further training a pre-trained model on new data.

[3]Standoff format: a method used in natural language processing (NLP) and text annotation where annotations are stored separately from the source text.

5. **Making Predictions:** After the model is trained, it is used to make predictions on the dataset. This is where the actual NER process occurs, as the model is now capable of recognizing named entities in the text.

6. **Saving Predictions and Training Process:** Finally, the script saves the predictions made by the model, as well as details of the training process.

The procedure for the BiLSTM-CRF above demonstrates the steps to take advantage of this neural approach for de-identification. This script effectively leverages the power of the BiLSTM-CRF method for de-identification by converting the input data into Flair sentences, generating embeddings, initializing or loading the BiLSTM-CRF model, training or fine-tuning, making predictions, and saving the results.

The method utilized by Deidentify draws insightful conclusions with the BiLSTM-CRF model. They argue that existing rule-based methods developed specifically for Dutch-language medical records do not generalize well to new domains. Contrarily, neural approaches to de-identification represented by the BiLSTM-CRF model have shown impressive results, showing excellent generalization performance across different languages and domains.

Essentially, the Deidentify method and its neural method BiLSTM-CRF play a key role in context. It provides an efficient, scalable and general de-identification solution that can be applied across different languages and medical domains. Therefore, it proposes an effective "out-of-the-box" health record de-identification method, which is crucial for further experiments and comparative analysis in the following sections.

3.3 Evaluation Metrics

The evaluation of Deduce and Deidentify de-identification methods involves a variety of key metrics Buitinck et al. (2013), which are clarified below through mathematical definitions and pseudocodes.

3.3.1 Metrics Definition

- **Precision (P):** The accuracy of positive predictions is calculated by calculating the proportion of true positive predictions to the total number of positive predictions, defined as:

$$P = \frac{TP}{TP + FP} \tag{1}$$

where TP = True Positives (the number of correctly identified sensitive entities) and FP = False Positives (the number of non-sensitive entities incorrectly identified as sensitive).

- **Recall (R):** Measures the proportion of actual positives that are correctly identified, i.e. the ratio of true positive predictions to the total number of actual positive instances:

$$R = \frac{TP}{TP + FN} \tag{2}$$

where $FN =$ False Negatives (the number of sensitive entities that were not identified).

- **F1 Score:** The F1 score is a measure of test accuracy - it is the harmonic mean of precision and recall. It has a maximum score of 1 (perfect precision and recall) and a minimum score of 0. The calculation formula is:

$$F1 = \frac{2 * P * R}{P + R} \tag{3}$$

- **Support:** Support is the number of times the class actually occurs in the specified class. It does not participate in the calculation of precision, recall or F1 score, but is used for weighted average:

$$Support = Number\ of\ actual\ occurrences\ of\ the\ label\ in\ the\ specified\ class \tag{4}$$

- **Macro Average:** The macro average is the average of the metrics for each class, regardless of class distribution/imbalance. It is calculated by summing the individual precision, recall or F1 score for each class and dividing by the number of classes, where N is the number of classes and $metric_i$ is the metric computed (eg, precision, recall, etc.) for i^{th} class:

$$Macro\ Avg\ (metric) = \frac{1}{N} \sum_{i=1}^{N} metric_i \tag{5}$$

- **Weighted Average:** We compute different types of weighted averages based on various metrics such as precision, recall or F1 score. All these calculations take into account the support of each class, i.e. the number of real instances of each class. The general formula for calculating the weighted average of any indicator is as follows:

$$Weighted\ Avg\ (metric) = \sum_{i=1}^{N} Support_i * Metric_i \tag{6}$$

In this equation, $Support_i$ is the support or actual number of occurrences of class i^{th} in the dataset, and $Metric_i$ is the calculated value of the selected metric (such as precision, recall or F1 score) for the i^{th} class. The sum covers all categories from 1 to N. This weighted average provides a single metric that takes into account the performance (given by a specific metric) and popularity (support) of each class in the dataset.

- **Accuracy:** Accuracy is the proportion of the total number of predictions that were correct. It is computed as the sum of true positives (TP_i) for each class divided by the

sum of true positives, false positives (FP_i), true negatives (TN_i), and false negatives (FN_i) for all classes.:

$$Accuracy = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + FP_i + TN_i + FN_i)} \tag{7}$$

3.3.2 Pseudocode for Evaluation

The pseudocode below demonstrates how the evaluation process is carried out:

```
Algorithm Evaluation_of_DeIdentification_Method

  Input: A .jsonl file containing labeled Dutch medical texts.
  Output: A classification report detailing the performance of the
  de-identification method.

  1. Initialize empty lists to hold the input texts and their corresponding
  true labels.
  2. Load data from the .jsonl file:
    - Append each text to the 'texts' list.
    - Append a dictionary of span labels to the 'true_labels' list, where
    each dictionary maps a (start, end) tuple to its label.
  3. Initialize an empty list to hold the predicted labels.
  4. For each text in 'texts':
    - De-identify the text using the object.
    - Append a dictionary of annotation labels to 'predicted_labels',
    where each dictionary maps a (start_char, end_char) tuple to its
    corresponding label.
  5. Flatten 'true_labels' and 'predicted_labels' into lists of label tuples.
  6. Sort 'true_labels_flat' and 'predicted_labels_flat' by their position
  tuples.
  7. Create a list 'true_labels_names' containing only the label names from
  'true_labels_flat'.
  8. Initialize an empty list 'predicted_labels_names_matched'.
  9. For each true_label in 'true_labels_flat':
    - If there exists a predicted_label in 'predicted_labels_flat' with the
    same position as true_label, append the name of predicted_label to
    'predicted_labels_names_matched'.
    - Otherwise, append 'No Prediction' to 'predicted_labels_names_matched'.
  10. Print a classification report comparing 'true_labels_names' and
  'predicted_labels_names_matched', handling the case of division by zero.

End Algorithm
```

This pseudocode[4] provides a procedural overview of the Python code for the evaluation process using Precision, Recall, and F1 Score. These metrics help evaluate the performance of the de-identification method under investigation.

[4]Evaluation pseudocode: this evaluation method is different from Deidentify, we create the evaluation process by getting the prediction from Deduce/Deidentify.

4 Results

This section discusses the results of our comparative analysis of Deduce and Deidentify on synthetic datasets and annotation-based datasets. We employ multiple metrics, including precision, recall, F1-score, and accuracy, to evaluate and compare their performance. Note that due to legal restrictions and various technical reasons, the evaluation code of the annotation-based dataset was run by Dr. Pablo Mosteiro Romero (UU) on the clinical notes of the Psychiatry Department at UMC Utrecht. His involvement ensured compliance with the necessary regulations and provided accurate insights into the performance of these models in a real-world clinical setting.

4.1 Performance Metrics

For ease of readability, performance metrics on synthetic datasets for Deduce and Deidentify are presented in Tables 3, for the annotation-based dataset are shown in Table 4.

	Deduce			Deidentify			
	P	R	F1	P	R	F1	S
DATUM	0.95	0.97	0.96	0.95	0.87	0.91	99
INSTELLING	0.93	0.38	0.54	0.76	0.63	0.69	100
LEEFTIJD	1.00	0.80	0.89	0.53	0.26	0.35	100
LOCATIE	0.90	0.52	0.66	0.87	0.65	0.74	100
PATIENTNUMMER	1.00	0.97	0.98	1.00	0.18	0.31	100
PERSOON	0.93	0.99	0.96	0.03	1.00	0.06	100
TELEFOONNUMMER	0.98	0.96	0.97	0.57	0.27	0.37	100
URL	0.98	0.98	0.98	0.97	0.34	0.50	100
accuracy			0.82			0.40	799
macro avg	0.85	0.84	0.77	0.67	0.52	0.39	799
weighted avg	0.96	0.82	0.87	0.83	0.40	0.48	799

Table 3: The evaluation result of Deduce and Deidentify performance on Synthetic Dataset

	Deduce			Deidentify			
	P	R	F1	P	R	F1	S
DATUM	0.94	0.80	0.86	0.97	0.86	0.91	724
INSTELLING	1.00	0.38	0.55	0.93	0.33	0.49	522
LEEFTIJD	1.00	0.26	0.41	0.96	0.27	0.42	98
LOCATIE	0.93	0.40	0.56	0.82	0.63	0.71	188
PATIENTNUMMER	0.03	0.03	0.03	0.37	0.80	0.50	40
PERSOON	0.98	0.79	0.87	0.20	0.77	0.32	271
TELEFOONNUMMER	0.96	0.80	0.87	0.83	0.91	0.87	55
URL	0.81	0.93	0.87	0.59	0.93	0.72	28
accuracy			0.66			0.46	2725
macro avg	0.74	0.60	0.56	0.67	0.65	0.49	2725
weighted avg	0.95	0.66	0.76	0.87	0.46	0.46	2725

Table 4: The evaluation results of Deduce and Deidentify performance on Annotation-based Dataset

4.2 Comparative Analysis

Our comparative analysis revealed differences in the performance of Deduce and Deidentify across different classes of identifiable information and in terms of overall accuracy.

4.2.1 Deduce

On the synthetic datasets, the Deduce model demonstrates a high F-1 score in recognizing and tagging elements such as DATUM, LEEFTIJD, PATIENTNUMMER, PERSOON, TELEFOONNUMMER and URL, indicating its ability to accurately identify these categories.

The limitation of the detection of Dutch institution tags "INSTELLING", which may be due to the fact that Deduce's institution names are mainly from psychiatric care institutions. On the other hand, the institution names of our comprehensive dataset cover all medical institutions in the Netherlands. In addition, there are various ways to express the name of the medical institution, such as different abbreviation formats or reversed order of the text before and after. This variability can pose challenges for Deduce to identify them correctly. To overcome this problem, enriching Deduce's data source of institution names is proposed. By integrating more diverse names of medical institutions across the Netherlands and employing flexible matching algorithms or natural language processing techniques, it should be possible to account for the different representation formats. Such measures may improve the accuracy of Deduce's detection of "INSTELLING" entities, regardless of their presentation format.

The low recall for the "LOCATIE" category could potentially be due to the fact that the data might not entirely align with our expectations. One possible reason could be the limitations of our Deduce system, which has predefined records for 2647 places of residence. In our Synthetic Dataset, geographical locations are complex and randomized, which means they might not always appear in the whitelist provided by Deduce. Furthermore, there might also be inconsistencies in the order of the street name, city name, and postcode. In some instances, the data might only contain the postcode and city without including the street and house number, further complicating the task. This complexity could lead to the observed lower recall rates.

Thus, despite Deduce accurately identifying certain categories, it frequently misses a majority of the actual instances of some entities ("INSTELLING", "LOCATIE"). This is primarily a result of the limitations of the dataset itself and Deduce is not designed diverse method for all the case. If the dataset is generated deeply following the blacklist and whitelist of Deduce, its ability to generalization in a real-world context cannot be truly tested, and the results may appear more positive than they actually are.

On the annotation-based datasets, this dataset is real-world data in which the data contains diverse expressions without the limitations of synthetic datasets, but it also means it becomes more complex. Deduce in DATUM, INSTELLING, LEEFTIJD, LOCATIE, PERSOON, TELEFOONNUMMER all show high precision, but PATIENTNUMMER has a surprisingly low precision rate, which means that Deduce does not recognize the correct phone number well

in the real-world data set.

A specific observation is related to how phone numbers are formatted. In some instances, phone numbers are written with the prefix separate from the rest of the number, for example: 06 12345678. Deduce often makes an error in these cases, tagging "1234567" as a patient number instead of recognizing it as part of a phone number. This underscores the nuances and challenges inherent in de-identifying real-world medical texts, where variations in data representation can impact the accuracy of de-identification tools like Deduce.

From the comparison of the results on the two datasets of Deduce, the accuracy of Deduce in real-world data is not as good as that of synthetic data. My analysis is: in real-world data, although there is no limitation as in synthetic data sets, the diverse expressions in real-world datasets affect the performance of Deduce, so the probability of Deduce identifying entities becomes low. Just like the results obtained from synthetic data sets, if the data sets are written in the normalized standard format, such as LEEFTIJD, PATIENTNUMMER, the F1 score will still remain high. It also proves that Deduce has good generalization ability in DATUM, TELEFOONNUMMER, URL in the real-world dataset.

4.2.2 Deidentify

On the synthetic datasets, conversely, the Deidentify method performs variable performance, with some classes showing low precision while others demonstrated high precision. Interestingly, the precision performance of Deidentify was relatively good, except for the INSTELLING and LEEFTIJD, PERSOON and TELEFOONNUMMER classes. However, Deduce outperformed Deidentify in recognizing INSTELLING, LEEFTIJD, LOCATIE, PERSOON, TELEFOONNUMMER, URL.

The recall performance of Deidentify was lower than that of Deduce for the DATUM, LEEFTIJD, PATIENTNUMMER, TELEFOONNUMMER, and URL classes. Despite Deidentify being a pretrained neural network model, its performance on the same synthetic datasets used by Deduce was not outstanding. Only in the case of INSTELLING, LOCATIE were its generalization ability superior to that of Deduce. Thus, the overall accuracy of Deidentify was lower than that of Deduce on synthetic datasets.

In terms of overall accuracy on the synthetic datasets, Deduce surpassed Deidentify. In other words, in the categories related to numbers, Deduce basically surpasses Deidentify. However, Deidentify demonstrated a better generalization ability in complex patterns such as INSTELLING, LEEFTIJD in unknown data.

On the annotation-based datasets, Deidentify has a good generalization ability for DATUM, LEEFTIJD, LOCATIE, TELEFOONNUMMER, but for the other categories is not high. Also, it can not identify most of the entities in the classes of INSTELLING and LEEFTIJD.

These results show the trade-offs involved in choosing a de-identification method for Dutch medical texts. Both Deduce and Deidentify has the generalization ability for unknown data. For the identification of fixed format data and categories related to numbers, the accuracy of Deidentify is not as good as that of Deduce; their difference on the synthetic dataset is 0.42. The performance on real-world datasets, the generalization ability of Deidentify is slightly higher than its performance on synthetic datasets, but still lower than Deduce of 0.2. So we can identify different entities of the dataset according to these two methods' advantages and disadvantages.

This comparative study highlights the need for task-specific customization and continuous improvement of de-identification techniques to ensure optimal results.

4.3 Error Analysis

We provide an extensive error analysis on Deduce and Deidentify on both datasets to identify why these methods make mistakes in the de-identification process and the potential method to reduce their errors. We are going to show the actual data fragment that the algorithm actually did and what it should have done.

4.3.1 Synthetic Dataset

1. **Deduce**
   Inside the DATUM category, for example, "3-11", Deduce only identifies "-11" as the date. Some typing errors in the address, such as "Boulevard 1945 3", Deduce incorrectly classifies "5 3" as the date. So Deduce should change or add strategy inside its regular expression to reduce the error result.

   For the INSTELLING class, Deduce sometimes can not identify their full name of them. For example, "OLVG West" is an institution name inside the true label, but actually, Deduce only finds "OLVG" as the institution label. This is not a big mistake of Deduce, but this does reduce the result of its performance. Besides, some institution name like "UMC+" is not inside the checklist, so Deduce only identifies "UMC".

   Some of the name inside PERSOON entity is also on the institution name list. For instance, in "Geert Grooteplein 10, 6525 GA Nijmegen", Deduce should use its street name patterns with the institution name list to identify "Geert Grooteplein" as location, but it actually labeled them as the person class.

   The age, like these two formats, "48 jaren oud", "een man van 38" cannot be labeled as LEEFTIJD. In our annotation strategy of the true label, only label the first two digits number as age. Deduce sometimes identifies the first two digits but sometimes identifies the whole phrase together as age. Thus Deduce needs to add some custom strategy inside their regular expression, to only label two digits numbers inside the age entity to increase the accuracy in identifying age.

LOCATIE entities have some typing errors inside the original dataset. But some street names and door numbers can not be detected by Deduce and are missed. For example, inside "Geert Grooteplein 10, 6525 GA Nijmegen" and "s-Gravendijkwal 230, 3015 CE Rotterdam", the "Geert Grooteplein 10" and "s-Gravendijkwal 230" are missing detected, they did not be stored into the trie-based hashing technique of Deduce.

The classes of PATIENTNUMMER and TELEFOONNUMMER are retrieved by 7 and 10 digits, respectively. The true label has typing errors like adding extra space and did not give the correct index, such as the true label showing "0612345678 " or " 0612345678". Also, the true label of the URL entity has the same typing error.

2. **Deidentify**
Missing this format "8/10" of date sometimes, and the incorrect identification of date, especially for example, "1081 HV Amsterdam", "1061 AE Amsterdam", their "1081" and "1061" will be identified as DATUM class by Deidentify.

   For INSTELLING entity, for instance, "VU Medisch Centrum", it is separately identified as "VU Medisch", "Centrum" which is not matched as the true label. And such as "1081 HV Amsterdam" identified as "HV Amsterdam", "8025 AB Zwolle" as "AB Zwolle". "Flevoziekenhuis" identified as "het Flevoziekenhuis", "Sint Lucas Andreas Ziekenhuis" as "Sint Lucas". So it seems the format with a space between two words, is easy to be recognized as the name of the institution.

   Inside the LEEFTIJD category, such as "48 jaren oud" will be detected as "48 jaren", or some labeled as, for example, "40-jarige", "32 jaar", etc., but in our annotation strategy, only the two-digit number, for instance, "48" is the correct result as age entity.

   For LOCATIE class, such as "De Boelelaan 1117, 1081 HV Amsterdam" which is only detected "De Boelelaan" part of the address as the location entity. Sometimes the rest of the address will be detected as an institution that such as "Dokter van Heesweg 2, 8025 AB Zwolle" is identified only "Dokter van Heesweg 2, 8025" as the location entity, and "AB Zwolle" is labeled as INSTELLING entity.

   In PERSOON entity, Deidentify misses labeling some person's name and detects an institution or street name as the person's name entity. For example, "Reinier de Graafweg 3" is a street address with a door number, but Deidentify identifies "Reinier" as the person's name. Also, it might detect only the first word of the person's name, such as "Max Roberts", Deidentify only identifies the first word "Max" as a person class.

   Most of the PATIENTNUMMER entity is missed by Deidentify. For TELEFOONNUMMER, Deidentify is also missing most telephone numbers inside the dataset, and it usually identifies the format of the telephone number as, for example, "t:0661275432", but the true label will only annotate with the actual ten digits number "0661275432", also it often identifies patient numbers such as "patnr:7895432" as telephone entity.

   Some URL categories were identified incorrectly by Deidentify because of the annotation error of the index inside the true label, but for most of the URLs Deidentify did not find them correctly.

4.3.2 Annotation-based Dataset

1. **Deduce**
   "onderweg" is a Dutch word meaning "underway". If used at the beginning of a sentence, it will be spelled "Onderweg". This looks similar to a street name beginning with a capital letter and ending with "weg". For example, in Utrecht, we have a Vleutenseweg. Deduce incorrectly tags this word as a location.

   "koos" is the past of "kiezen" (to choose). If used at the beginning of a sentence, it will be spelled "Koos". This is also a proper name. For example: $https://en.wikipedia.org/wiki/Koos\_Andriessen$. Deduce incorrectly tags this word as a person. The same happens with "Toon" or "Mooi", because 'koos', 'Toon' are inside the first names list, 'Mooi' is inside the surnames list.

   Phone numbers are sometimes written with the prefix separate from the rest of the number, for example: 06 12345678. Deduce incorrectly tags "1234567" as a patient number.

   "thuis" is a Dutch word meaning "at home". If used at the beginning of a sentence, it will be spelled "Thuis". This looks similar to some Dutch names, though it isn't a name, Deduce labeled this as a person, because 'Thuis' is inside the surnames lookup list.

   Deduce chooses to annotate the title (like "mw", meaning Mrs) together with the name of the person. But in our Annotation-based dataset, we did not do that. The same thing happens with some dates versus date+time, or date+day-of-week.

   Some institutions have names that are also common words, like "karakter". They get incorrectly labeled by Deduce as institutions because it has been collected into the institutions' checklist.

   The Deduce tool demonstrates challenges in correctly tagging Dutch words due to similarities in spelling, particularly at the beginning of sentences. This results in issues like mistagging "onderweg" as a location, or common names like "Koos" and "Thuis" as persons. Additionally, Deduce misinterprets certain numerical sequences as patient numbers and joins titles with names, deviating from the practice in our Annotation-based dataset. There's also the complication of institutional names that are common words, leading to further mislabeling. These discrepancies highlight the need for refinements in the tool's annotative logic for the Dutch language.

2. **Deidentify**
   "Canada" is labeled by Deidentify as a location. As stated in the Deduce paper, the local guidelines don't require anonymizing the names of countries. Deidentify was trained on a different dataset, where locations might have been anonymized.

   "Koos" is labeled by Deidentify as a person, similar to Deduce, because this word can be both a verb and a name, And they identified it as a person's name.

   "feb" is labeled by Deidentify as a date. As stated in the Deduce paper, "only combinations of days and months are regarded as potentially identifying information". Thus,

"feb" alone should not be annotated.

The sequence "br><br" is labeled by Deidentify as a person. This does not make sense and it can be considered an error. To be fair, however, this sequence should probably not have been included in the source text.

Also, the sequence "br>2" is incorrectly labeled by Deidentify as a URL.

Note furthermore that all PATIENT instances are correctly labeled by Deidentify as a person. Deidentify does not have a patient category, so labeling it as a person should be considered correct.

The sequence "AT" is labeled by Deidentify as a person. This might be correct, or it might refer to an acronym. To elucidate this, we'd have to consult with domain experts.

Deidentify's labeling of various terms diverges from the annotation strategy and the insights from the Deduce paper. Like "Canada" is correctly labeled as a location but isn't required to be anonymized inside Deduce. Both "Koos" and "feb" are labeled as a person and date, respectively, with the former aligning with Deduce's understanding due to its dual meaning, while the latter shouldn't be annotated as a standalone month. "br><br" being labeled as a person is an error, and "br>2" being labeled as a URL is also incorrect. The labeling of "AT" remains ambiguous, warranting expert consultation. Thus Deidentify has the problem originally trained with its own guidelines and after its own training process, it still has problems detecting some entities correctly.

5 Conclusions

This thesis undertook a comparative study of Deduce and Deidentify de-identification techniques within the context of Dutch medical texts, aiming to answer the research question: 'How do the rule-based and neural network approaches to de-identification compare in terms of effectiveness, adaptability, and generalization when applied to Dutch medical data, and how can the strengths of these methodologies be used to improve the de-identification process and enhance patient privacy protection?'

5.1 Summary

Our experiments revealed that both Deduce and Deidentify have strengths and limitations, demonstrating variable effectiveness, adaptability, and applicability to Dutch medical data. In fixed-format synthetic data, Deduce surpasses Deidentify in terms of effectiveness, achieving a higher F1 score for categories like DATUM, LEEFTIJD, PATIENTNUMMER, PERSOON, TELEFOONNUMMER, and URL. Its rule-based approach, while requiring continuous rule modification and updating, provides a more precise and effective method for de-identifying structured data and numeric categories.

On the other hand, Deidentify's neural network approach demonstrates better adaptability and generalization capability when handling complex patterns like DATUM, LOCATIE on unknown or real-world data. Despite its relatively lower accuracy on synthetic datasets, it proves to increase when confronted with diverse and unstructured data, as reflected by its higher overall accuracy on the annotation-based dataset.

Thus, it is clear that each approach has its unique advantages. The rule-based Deduce performs better on structured and numeric data, while the neural network-based Deidentify excels in dealing with complex and unstructured patterns. This realization brings us closer to answering the latter part of our research question.

5.2 Discussion

In other words, when facing real-world data, after verifying the true labels of the dataset with accurate format checks and ensuring no errors, for entity categories such as INSTELLING, PA-

TIENTNUMMER, PERSOON, TELEFOONNUMMER, and URL, the Deduce method should be employed for de-identification. However, before utilizing Deduce, it is essential to heed the recommendations from the error analysis section and consult with Dutch language domain experts. Specifically, the lookup list, trie-based hashing, and regular expressions should be appropriately modified to enhance Deduce's accuracy. For the other entity categories, the Deidentify method can be considered for de-identification. Importantly, before using Deidentify, it would be beneficial to fine-tune the pre-trained model of Deidentify on the required dataset. Doing so can improve Deidentify's accuracy on that specific dataset.

By utilizing the accuracy of Deduce in de-identifying structured and numeric categories with Deidentify's adaptability and generalization ability for complex and unstructured data, the de-identification techniques' effectiveness, adaptability, and generalization on Dutch medical text can be improved. Thus we show the summary of the recommended de-identification methods for different PHI categories in Table 5.

PHI Category	**Deduce**	**Deidentify**
DATUM		✓
INSTELLING	✓	
LEEFTIJD		✓
LOCATIE		✓
PATIENTNUMMER	✓	
PERSOON	✓	
TELEFOONNUMMER	✓	
URL	✓	

Table 5: The Summary of Recommended De-identification Methods for Different PHI Categories

In essence, this study sheds light on the distinct strengths of Deduce and Deidentify. However, the full spectrum of their comparative efficacies, can only be deeply understood with broader access to the real-world datasets. Consequently, while the findings guide the current best practices in Dutch medical text de-identification, there remains a window of opportunity for further research, especially with access to richer and diverse real-world datasets.

5.3 Future Work

In view of the findings and the experiment result, the following potential step has been identified for future research on the De-identification of Dutch medical text:

1. **Expanding the dataset:** Experiment with a larger and more diverse dataset is necessary to improve Deduce and Deidentify's performance and generalization ability. This includes the identification of non-Dutch and non-EU institutions, various age representations, and non-Dutch names.

2. **Enhanced techniques:** Exploring other potential de-identification methods that offer superior precision, recall, and F1 score results is important. This can be realized by investigating emerging techniques, including large multi-language architectures (like GPT-4) related to Named Entity Recognition (NER) tasks.

3. **Improved generation of synthetic Dutch medical data:** The research highlighted the need for more robust, rich and diverse Dutch medical data. Therefore, future work can be focused on generating synthetic data that accurately correspond to real-world medical data's diverse and complex nature.

4. **Continued development and integration:** Collaborating with organizations like Nedap to further upgrade the current Deduce (1.0) model inside the Deidentify Python library would be beneficial. This collaboration would facilitate the integration of the latest version of Deduce (2.0), leveraging advancements in this field and improving the overall function of the library.

References

Aberdeen, J., Bayer, S., Yeniterzi, R., Wellner, B., Clark, C., Hanauer, D., Malin, B., and Hirschman, L. (2010). The mitre identification scrubber toolkit: Design, training, and assessment. *International Journal of Medical Informatics*, 79(12):849–859.

Agnikula Kshatriya, B. S., Sagheb, E., Wi, C.-I., Yoon, J., Seol, H. Y., Juhn, Y., and Sohn, S. (2021). Identification of asthma control factor in clinical notes using a hybrid deep learning model. *BMC Medical Informatics and Decision Making*, 21(7):272.

Ahmed, T., Aziz, M. M. A., and Mohammed, N. (2020). De-identification of electronic health record using neural network. *Scientific Reports*, 10(1):18600.

Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., van den Driessche, G., Lespiau, J., Damoc, B., Clark, A., de Las Casas, D., Guy, A., Menick, J., Ring, R., Hennigan, T., Huang, S., Maggiore, L., Jones, C., Cassirer, A., Brock, A., Paganini, M., Irving, G., Vinyals, O., Osindero, S., Simonyan, K., Rae, J. W., Elsen, E., and Sifre, L. (2021). Improving language models by retrieving from trillions of tokens. *CoRR*, abs/2112.04426.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Cai, J. and Strube, M. (2010). Evaluation metrics for end-to-end coreference resolution systems. In *Proceedings of the SIGDIAL 2010 Conference*, pages 28–36.

Dernoncourt, F., Lee, J. Y., Uzuner, Ö., and Szolovits, P. (2016). De-identification of patient notes with recurrent neural networks. *CoRR*, abs/1606.03475.

Douglass, M., Cliffford, G., Reisner, A., Long, W., Moody, G., and Mark, R. (2005). De-identification algorithm for free-text nursing notes. In *Computers in Cardiology, 2005*, pages 331–334.

Edemekong, P. F., Annamaraju, P., and Haydel, M. J. (2022). *Health Insurance Portability and Accountability Act*. StatPearls Publishing, Treasure Island (FL).

Ferrández, O., South, B. R., Shen, S., Friedlin, F. J., Samore, M. H., and Meystre, S. M. (2012). Evaluating current automatic de-identification methods with veteran's health administration clinical documents. *BMC Medical Research Methodology*, 12(1):109.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Islam, R. U., Hossain, M. S., and Andersson, K. (2020). A deep learning inspired belief rule-based expert system. *IEEE Access*, 8:190637–190651.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., and Levy, O. (2019). Spanbert: Improving pre-training by representing and predicting spans. *CoRR*, abs/1907.10529.

Khin, K., Burckhardt, P., and Padman, R. (2018). A deep learning architecture for de-identification of patient notes: Implementation and evaluation. *CoRR*, abs/1810.01570.

Menger, V., Scheepers, F., van Wijk, L. M., and Spruit, M. (2018). Deduce: A pattern matching method for automatic de-identification of dutch medical text. *Telematics and Informatics*, 35(4):727–736.

Meystre, S. M., Óscar Ferrández, Friedlin, F. J., South, B. R., Shen, S., and Samore, M. H. (2014). Text de-identification for privacy protection: A study of its impact on clinical text information content. *Journal of Biomedical Informatics*, 50:142–150. Special Issue on Informatics Methods in Medical Privacy.

Morrison, F. P., Li, L., Lai, A. M., and Hripcsak, G. (2009). Repurposing the Clinical Record: Can an Existing Natural Language Processing System De-identify Clinical Notes? *Journal of the American Medical Informatics Association*, 16(1):37–39.

OpenAI (2023). Gpt-4 technical report.

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., and Manning, C. D. (2020). Stanza: A python natural language processing toolkit for many human languages. *CoRR*, abs/2003.07082.

Stubbs, A., Kotfila, C., and Özlem Uzuner (2015). Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/uthealth shared task track 1. *Journal of Biomedical Informatics*, 58:S11–S19. Supplement: Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data.

Tawfik, N. S. and Spruit, M. R. (2019). Towards recognition of textual entailment in the biomedical domain. In *Natural Language Processing and Information Systems: 24th International Conference on Applications of Natural Language to Information Systems, NLDB 2019, Salford, UK, June 26–28, 2019, Proceedings 24*, pages 368–375. Springer.

TDSLAb (2023). Deduce: De-identification method for dutch medical text (demo) (version 1.0.8) [screenshot]. https://tdslab.nl/deduce#results.

Tikk, D. and Solt, I. (2010). Improving textual medication extraction using combined conditional random fields and rule-based systems. *Journal of the American Medical Informatics Association*, 17(5):540–544.

Trienes, J., Trieschnigg, D., Seifert, C., and Hiemstra, D. (2020). Comparing rule-based, feature-based and deep neural methods for de-identification of dutch medical records. *arXiv preprint arXiv:2001.05714*.

Uzuner, O., Luo, Y., and Szolovits, P. (2007). Evaluating the State-of-the-Art in Automatic De-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

van Haastrecht, M., Sarhan, I., Yigit Ozkan, B., Brinkhuis, M., and Spruit, M. (2021). Symbals: A systematic review methodology blending active learning and snowballing. *Frontiers in research metrics and analytics*, 6:685591.

Voigt, P. and von dem Bussche, A. (2017). *The EU General Data Protection Regulation (GDPR)*. Springer International Publishing.

Wang, Q., Zhou, Y., Ruan, T., Gao, D., Xia, Y., and He, P. (2019). Incorporating dictionaries into deep neural networks for the chinese clinical named entity recognition. *Journal of Biomedical Informatics*, 92:103133.

Yang, H. and Garibaldi, J. M. (2015). Automatic detection of protected health information from clinic narratives. *Journal of Biomedical Informatics*, 58:S30–S38. Supplement: Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data.