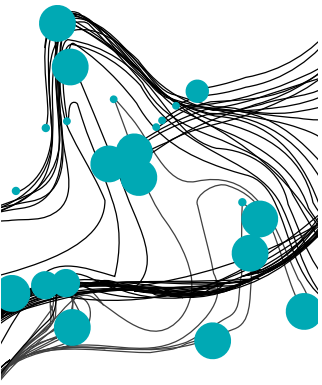
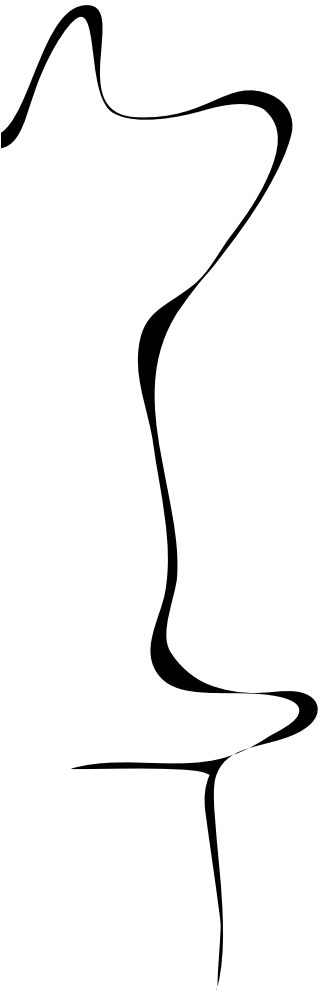


.92155

DMB

DATA MANAGEMENT AND BIOMETRICS



DEVELOPMENT OF A MODEL TO EXTRACT DIABETES TYPE AND YEAR OF DIAGNOSIS FROM MEDICAL TEXTS

Kenzy Dario Sanjaya

MASTER'S ASSIGNMENT

Committee:
dr. A. Briassouli PhD
dr. F. Ahmed
J.J. van de Beld MSc

January, 2026

Data Management and Biometrics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Contents

Chapter 1

Introduction

Diabetes is one of the most common chronic illnesses in the Netherlands and requires continuous monitoring and long-term management. Accurate knowledge of a patient's diabetes type and the year of diagnosis is crucial for selecting appropriate treatment strategies, assessing disease progression, and evaluating long term risks. Although electronic health records (EHRs) include structured fields related to diabetes, essential details, particularly the initial diagnosis, are often recorded only in free text clinical notes. These narrative documents vary widely in terminology, writing style, and completeness, making retrieval of clinically relevant information difficult.

Inaccurate or incomplete diabetes documentation can negatively affect both day-to-day clinical decision making and secondary use of hospital data. For example, if a patient with type 1 diabetes is incorrectly registered as having type 2 diabetes, insulin therapy decisions or complication risk assessments could be affected. Likewise, missing or inconsistent diagnosis dates complicate longitudinal analyses and cohort selection for research. With the increasing volume of clinical documentation, manually reviewing patient histories is no longer scalable and is prone to human error. Medical staff may not have the time or resources to search through lengthy reports, and contradictory statements across clinical encounters introduce additional challenges.

Natural Language Processing (NLP) offers promising solutions for extracting structured information from unstructured clinical text. Numerous studies have demonstrated that NLP techniques, including Named Entity Recognition (NER) and temporal information extraction can outperform manual review by improving consistency, resolving ambiguities, and identifying temporal relationships in clinical records [7, 3, 4]. Despite these advancements, research on NLP for extracting diabetes-related information from Dutch clinical text remains limited.

This thesis, conducted in collaboration with Ziekenhuisgroep Twente (ZGT), aims to automatically extract diabetes type and year of diagnosis from unstructured Dutch EHR documents. Since the start of the project, a literature review has been completed focusing on clinical information extraction, NER approaches, temporal normalization, and prompt-based techniques using instruction-tuned language models. Preliminary experiments using MedRoBERTa.nl and several local language models have been performed to evaluate feasibility for Dutch clinical text.

Building on these findings, two complementary extraction approaches are developed and evaluated in this thesis: (i) a NER-based pipeline using MedRoBERTa.nl combined with rule-based temporal normalization, and (ii) a prompt-based extraction approach using locally deployable instruction tuned language models. Both methods are implemented and tested on anonymised clinical data provided by ZGT, with the goal of assessing extraction accuracy, robustness, and applicability within a real hospital setting.

Research Questions

- **RQ1:** To what extent can NLP be used to reliably extract the diabetes type and year of diagnosis from unstructured Dutch EHR text, while resolving conflicting or ambiguous information?
- **RQ1.1:** Which approach is more effective for extracting diabetes diagnosis year and type: a rule-based NER pipeline or a prompt-based LLM approach?
- **RQ1.2:** Which clinical document types provide the most reliable and complete information regarding the initial diabetes diagnosis?

Chapter 2

Related Work

Natural Language Processing (NLP) has become an essential tool in modern clinical research, one of the most useful appplication of NLP is the transformation of unstructured medical text into structured, machine readable data. Clinical reports such as diagnostic dates, progress notes, and diagnostic reports often contain important information that is not available in structured fields of electronic health records (EHRs).

Over the past two decades, various approaches have been proposed for clinical information extraction, from rule based systems and classical machine learning algorithms to deep learning approaches, such as transformer based language models [7, 11]. These systems are typically designed to identify named entities, classify clinical events, and detect temporal information.

This chapter reviews existing work in clinical NLP that informs the development of the proposed system. The focus is on methods for diagnosis date extraction, the use of large language models for temporal reasoning, and domain specific adaptations for Dutch clinical text.

2.1 Extracting and Classifying Diagnosis Dates from Clinical Notes

Fu et al. [4] present a case study with goal to extract and classify diagnosis dates from unstructured clinical notes of patients with myeloproliferative neoplasms (MPNs). In this paper, they developed and evaluated a pipeline combining lightweight NLP tools and regular expressions to identify relevant temporal expressions because the dates are not consistently written down in a structured way.

There are three methods for date extraction that were compared: `parsedatetime`, `spaCy`, and a custom regular expression based extractor adapted from `timex.py`.

The regex based method outperformed the others, correctly matching 83.0% of manually annotated diagnosis dates, whereas spaCy and parsedatetime fell behind, especially on relative and partial date formats. This lower performance was due to limitations in their general purpose design: spaCy often missed dates presented outside of narrative text (such as lists or metadata), while parsedatetime tended to misinterpret relative or partial dates, frequently extracting future dates instead of past events as found in clinical notes.

The authors concluded the importance of task specific temporal extraction rules in clinical NLP.

2.2 Large Language Models for Temporal Information Extraction

Zhang et al. [13] evaluated the capability of two large language models, ChatGPT (GPT-4) and LlaMA-2 in extracting structured clinical information, specifically cognitive test scores (MMSE and CDR) and their associated dates, from unstructured clinical notes. Their dataset consisted of over 34,000 clinical notes filtered for mentions of cognitive assessments. A subset of 765 notes was manually annotated by 22 medical experts and used to assess LLM performance.

In their research, ChatGPT shows high extraction accuracy (83% for MMSE, 89% for CDR), with excellent recall and low hallucination rates, outperforming the open source LlaMA-2 model. The study demonstrated that with careful prompt design and preprocessing, LLMs can extract time stamped clinical information with high reliability.

Their findings shows the importance of human validation and how LLMs can reduce the need for rule based systems or extensive training data when extracting structured data such as diagnosis dates from free text notes.

This work is relevant to the current proposed solution, as it validates prompt based approaches to temporal information extraction and supports the viability of large language models for data sensitive tasks when privacy based infrastructure (e.g. local deployment) is used.

2.3 Dutch Clinical Information Extraction using BERT

Muizelaar et al. [9] explored the classification of patient lifestyle characteristics. Specifically smoking, alcohol, and drug usage from Dutch clinical texts using a variety of methods, including string matching, classical machine learning, and BERT based models. The study involved over 148,000 anonymized clinical notes from HagaZiekenhuis and introduced both automatically labelled and manually annotated datasets.

Among the models evaluated, a further pretrained version of MedRoBERTa.nl (MedRoBERTa.nl-HAGA) achieved the highest performance on smoking (Macro F1 = 0.93) and drug usage (Macro F1 = 0.77), outperforming string matching and traditional classifiers. Notably, ClinicalBERT fine tuned on English translated Dutch texts performed best on alcohol classification (Macro F1 = 0.80), suggesting that translation can be a viable strategy for leveraging large English clinical models in lower resource language settings.

2.4 Takeaways From Related Works

The reviewed studies highlight several insights relevant to the current thesis. First, Fu et al. [4] demonstrate the importance of task specific temporal extraction rules, suggesting that a rule based component will likely be necessary for handling ambiguous or inconsistent diagnosis years in Dutch EHRs. Second, Zhang et al. [13] show that prompt based large language models can achieve high accuracy in extracting temporal information, indicating that such approaches may complement or even replace traditional NER pipelines when deployed in privacy preserving local environments. Finally, Muizelaar et al. [9] confirm the effectiveness of domain adapted Dutch language models such as MedRoBERTa.nl.

Together, these findings justify the dual approach proposed in this work: (1) a MedRoBERTa based NER model with rule based temporal normalization, and (2) a prompt based extraction pipeline using local language models.

Chapter 3

Dataset

3.1 Research Environment

This thesis is conducted in collaboration with Ziekenhuisgroep Twente (ZGT), a large regional hospital organization in the eastern Netherlands with locations in Almelo and Hengelo. ZGT Hengelo, where this research is primarily based, provides a wide range of medical services including internal medicine, endocrinology, and specialized diabetes care. The hospital serves a diverse patient population and maintains a strong focus on multidisciplinary collaboration across departments.

ZGT treats a substantial number of patients with diabetes each year. Approximately 2,500 patients with type 1 diabetes receive care at ZGT, typically attending at least three outpatient visits per year. While most patients with type 2 diabetes in the Netherlands are managed by general practitioners, many hospitalized patients with diabetes at ZGT have type 2 diabetes. This mix of outpatient and inpatient documentation results in a broad variety of clinical notes relevant to diabetes management.

For this research, three types of clinical documents are particularly important: Naslag, Voorblad, and Zorgdomein. Naslag and Voorblad reports were provided in an already extracted and tabular form, making them semi-structured data sources. Zorgdomein documents, in contrast, were available only as PDF files and therefore represent fully unstructured clinical text. In total, approximately 70% of the available dataset consists of unstructured text originating from Zorgdomein, highlighting the need for automated extraction methods capable of handling varied and inconsistent documentation formats.

3.2 Provided Data

The provided data of the patient consists of Voorblad (Electronic Health Record), Naslag (Clinician's note), and Zorgdomein (Letters from and to General Pracitioners). patients have at least one of these documents and potentially all documents. The extraction result points to where the discovery is found manually. The total manually annotated data is 460 patient, with the diagnoses being the patient number, study, diagnosis type, where is the type found, diagnosis date, and where is the date found.

3.2.1 Voorblad

Voorblad is the summary or overview of the EHR, containing a concise overview of the patient's medical condition. The Voorblad data is the result of extraction by ZGT Hengelo and is provided in a Comma Separated Values (CSV) format. An example of the extracted data is shown in Table 3.1.

Although the Voorblad contains a summary of the patient's EHR, it does not necessarily include the type and date of diabetes diagnoses. For instance, it may only indicate that the patient has diabetes without specifying the type or the date. Relevant sections in the Voorblad are *Relevante Voorgeschiedenis*, *Overige Voorgeschiedenis*, and *Active Diagnoses*. These relevant sections are already the data that is extracted to be used in this research.

patientnr	datum	diagrel	code	specialism	omschrijving
P005	1986-01-01	V	INT001	INT	Diabetes mellitus type X, sinds 1995 insuline pomp therapie
P005	1995-01-01	V	INT002	INT	Laparoscopische sterilisatie
P005	1999-01-01	V	INT003	INT	CTS rechts
P005	2001-01-01	V	INT004	INT	Buikklachten obv obstipatie
P005	2011-01-01	V	INT005	INT	Chronische primaire insomnia
P005	2012-01-01	V	INT006	INT	Subklinische hypothyreoidie
P005	2015-10-27	D	DBC001	INT	Diabetes mellitus chronisch pomptherapie @DBC
P005	2016-07-21	D	DBC002	ORT	Tendinitis van heup
P005	2017-08-02	D	DBC003	DER	Alopecia telogeen effluvium
P005	2019-06-12	D	DBC004	CHI	Fractuur van femurhals links
P005	2019-06-12	D	DBC005	CHI	Fractuur van femurhals rechts
P005	2020-07-03	D	DBC006	CHI	NULL

TABLE 3.1: Illustration of anonymized patient voorblad record with multiple diagnoses over time

patientnr	text	date	time	type_of_text	text_code	specialism_filepart_text	docters_code	docters_name
P002	\rtf1\ansi Diabetes mellitus type 1; Inreda AP	2022-02-02	09:21	Reden van komst / Verwijzing	C0001	DIA	D1234	Dr. A. Example

TABLE 3.2: Anonymized Naslag record

3.2.2 Naslag

Naslag is the clinician's note when a visit happened. It is a summary of when a patient visits the clinic. Typically this happens minimum two times per year.

The naslag data is the result of an extraction from ZGT Hengelo. This means that the format is not as human readable. For example the data can still be "dirty" as in it still has traces of extraction code result. This can be seen in the table 3.2 in the column Text. Typically, it contains reason for coming and summary usually it's only for the type of diabetes. However, parts of the text still include leftover extraction artifacts such as \rtf1\ansi, which are traces from the original RTF formatting of the EHR and not meaningful clinical content.

3.2.3 Zorgdomein

patientnr	doc_type	doc_id	doc_date	section	text
P001	zorgdomein	D001	2017-10-09	Episodelijst	15-05-2016, chronische aandoening X behandeld met medicatie Y, A12.34. 02-02-1990, metabole aandoening Z, specialist, T56.78
P001	zorgdomein	D001	2017-10-09	Probleemlijst	10-09-2017, klacht A (bijv. pijn gewricht), L45. 05-09-2017, klacht B (bijv. spierpijn), L67. 20-08-2017, klacht C (bijv. huidirritatie), S34. 11-06-2016, klacht D (bijv. wond arm), S92. 03-06-2015, klacht E (bijv. mobiliteitsprobleem), L23. 15-05-2011, klacht F (bijv. slaapprobleem, medicatie X voorgeschreven), P07. 07-03-2005, algemeen journaal, A88.

TABLE 3.3: Example of extracted data from a Zorgdomein Report

Zorgdomein are letters that are exchanged between general practitioners (GPs) and other healthcare providers. From these letters, only two sections are meaningful in the context of this research: *Probleemlijst* and *Episodelijst*. The *Probleemlijst* provides a summary of the patient's main ongoing health conditions, while the *Episodelijst* lists specific medical episodes or events over time, including dates and the nature of each complaint or diagnosis (e.g. pain in the hand or foot, skin rash, neuropathy). A single patient can have multiple Zorgdomein documents from their GP, each potentially containing updated information.

The table 3.3 is an extracted and anonymized representation of the PDF version of such a document. Only the *Probleemlijst* and *Episodelijst* sections are extracted, since these contain the

relevant information needed for this research specifically, the type of diabetes and the date of diagnosis.

3.2.4 Diabase and Dialect

Diabase and Dialect refers to the study the patients is refering to. Most of the Diabase patient has Diabetes type 1 while most of the Dialect patients has Diabetes type 2.

3.3 Manual Extraction Result

patientnr	study	diagnosis	diag_doc_name	diag_doc_section	date_diagnosis	date_doc_name	date_doc_section	remarks
P004	STUDYX	type 1	Document A (20/6/2021): Zorgdomein ZD000001	Probleemlijst 10-05-2008, Cardiale afwijking, hypertensie, K11.01 15-02-2001, Diabetes Mellitus Type 1, T90.XX 07-07-1999, Schildklieraandoening, T81.XX	15/02/2001	Document A (20/6/2021): Zorgdomein ZD000001	Probleemlijst 10-05-2008, Cardiale afwijking, hypertensie, K11.01 15-02-2001, Diabetes Mellitus Type 1, T90.XX 07-07-1999, Schildklieraandoening, T81.XX	NA

TABLE 3.4: Anonymized example of manually annotated patient record

In the table 3.4 is the example of a manual annotation by ZGT Hengelo. It was stated that the diagnosis is the study is Diabase with type 1 diabetes with the date of diagnosis being 15/02/2001. Both the type and date diagnosis were found in Zorgdomein document.

3.4 Expert Feedback and Clinical Observations

To validate the relevance of the problem and understand its real world implications, We gathered feedback was gathered from hospital staff at ZGT Hengelo. Their observations notes some critical limitations in the current clinical workflow for determining diabetes diagnosis details from Electronic Health Records (EHRs).

One recurring issue is the high frequency of diagnosis years recorded as either **2015/2016** or **1900**, in Voorblad. Such issues in EHR fields have showed the need for the development of specialized extraction approaches [4]. The year 2015/2016 appears disproportionately accounting for over 50% of extracted diagnosis dates, this is due to a change in the EHR system around that period, which makes the year of diagnosis to become 2015/2016. The year 1900 is often entered as a placeholder when the actual year of diagnosis is unknown. Both cases lead to unreliable data and make the validity of clinical and epidemiological research unreliable.

Misidentifying the year of diagnosis affects not only research integrity but also patient care. For example, incorrectly assuming a recent diagnosis could influence treatment plans or eligibility for certain care programs. On the other hand, misclassification of the diabetes type while less frequent poses a significant clinical risk. Type 1 diabetes always requires insulin therapy, whereas Type 2 may not. An incorrect assumption about the type can therefore lead to inappropriate treatment and serious health consequences.

Clinicians also reported that they often rely solely on the *Relevante Voorgeschiedenis* (Relevant Medical History) section, as it is the quickest to review during patient visits. However, this section does not always contain the most reliable or complete information. Other sections, such as *Actieve Diagnoses* or *Verwijzingen*, may include more accurate diagnosis details but are often overlooked.

From these observations, it becomes clear that an automated and reliable method is needed to extract both the diagnosis year and type from full EHR text. Beyond supporting daily clinical decision making, such a system could also improve the quality of clinical research by providing cleaner and more consistent data. For example, correcting placeholder years such as 1900 or ambiguous ranges like 2015/2016 would increase the validity of studies and analyses. Therefore, resolving inconsistencies and assessing the trustworthiness of different document sections offers a the following benefits: clinical utility and long term scientific reliability. In the following, we analyze how text data can be processed and structured in order to correctly extract diagnosis years and types, which directly connects to the methods discussed in the next section.

3.5 Dataset Overview

The dataset used in this research contains three main clinical document types from ZGT: *Voorblad*, *Naslag*, and *Zorgdomein*. Before developing automated extraction models, the first step in this project was to obtain a high quality ground truth dataset manually annotated by a healthcare professional at ZGT.

The annotation process was performed manually by a trained healthcare professional at ZGT. Each clinical document for a given patient was read in full, and the annotator extracted the following elements: the diabetes type, the diagnosis year, and the specific document containing the diabetes type and the diagnosis year. The source of the diabetes type and diagnosis year might be different or the same, depending on where the annotator found it. All annotations were recorded in a structured CSV format. This really helps with a consistent, high quality ground truth to evaluate automated extraction models.

Since this manual extraction is considered the most reliable reference, the distribution of where clinicians found the relevant information serves as an important indicator of document usefulness and extraction difficulty.

Manual Labelling Occurrence Counts

The following counts show how often the healthcare professional found the diabetes **type** and **diagnosis year** in each document type:

- **diag_doc_name** (where diabetes *type* was found):
  - Voorblad: 299
  - Naslag: 40
  - Zorgdomein: 249
- **date_doc_name** (where the *diagnosis date* was found):
  - Voorblad: 283
  - Naslag: 17
  - Zorgdomein: 287

These counts represent the trusted, clinician verified annotations that later serve as ground truth for evaluating automated extraction systems.

Manual Annotation Results

Figures 3.1 and 3.2 show the distribution of where clinicians found relevant information during manual labelling.

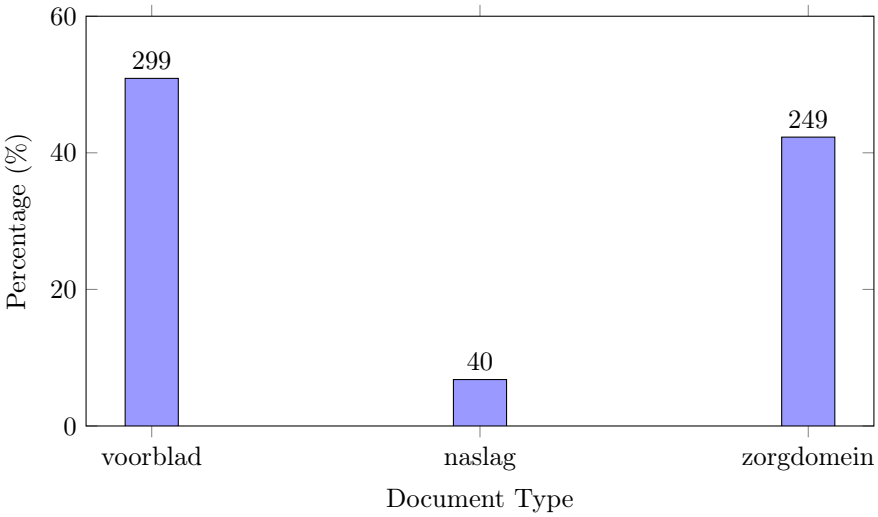


FIGURE 3.1: Relative frequency of documents containing diabetes type information (`diag_doc_name`), with absolute counts shown on the bars.

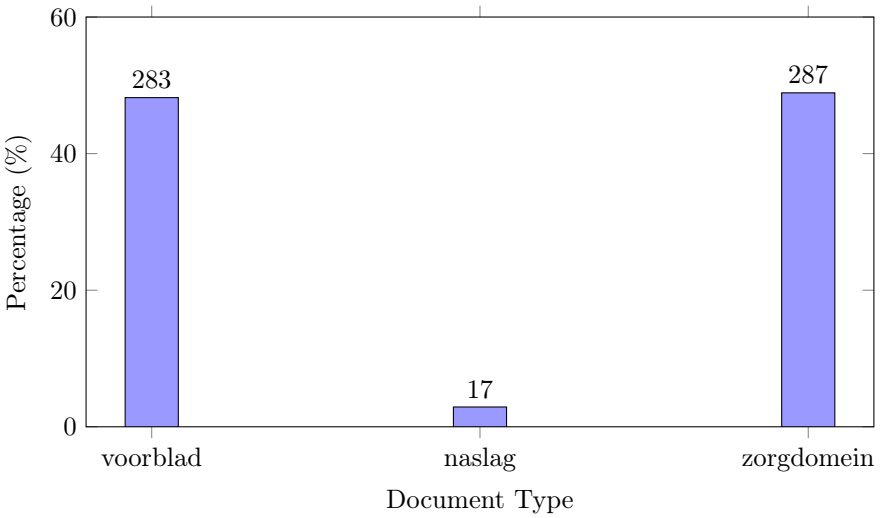


FIGURE 3.2: Relative frequency of documents containing diagnosis date information (`date_doc_name`), with absolute counts shown on the bars.

Insights From Manual Annotation

The manual counts reveal important characteristics of the dataset and provide insight into how different document types are actually used in clinical practice. Voorblad is the summary or overview of the EHR, containing a concise overview of the patient's medical condition. The Voorblad document is typically the document that caregivers consult first, as it is one of the most structured and generally up-to-date summaries of the patient's medical history. However, it may still contain incomplete or outdated information, and diagnosis years may occasionally be represented using placeholders such as 1900 or 2015/2016 due to historical system transitions or missing data. Despite these issues, the Voorblad remains a valuable and interpretable source and is often trusted unless contradictory evidence is found elsewhere.

In contrast, Naslag documents contain very few usable mentions. Naslag entries tend to be lengthy, noisy, and include substantial irrelevant informations, making them difficult to find relevant information here. This aligns with the low number of manual detections attributed to Naslag in the ground truth.

Multimedia documents, or Zorgdomein were described as letters to and from general practitioners, containing information that is often highly similar in quality to the Voorblad. As a result, they often restate key diagnoses in a clean and reliable manner, which explains why manual annotation found them to be a frequent and trustworthy source of diabetes type and diagnosis year.

Caregivers also provided an important practical rule for resolving conflicting information across documents. When multiple entries refer to a patient's diabetes type or diagnosis year, the value that appears most frequently is typically the correct one. For example, if two documents state "type 2" and one states "type 1," clinicians would interpret the correct diagnosis as type 2. This reflects the clinical intuition that repeated documentation across sources increases confidence in the accuracy of a finding. These insights guide both the interpretation of the dataset and the design of automated extraction methods.

3.6 Data Cleaning

Before training and evaluating NLP models, the clinical dataset underwent a series of data cleaning and preprocessing steps to ensure consistency, remove noise, and prepare the text for token level annotation. This process included standardizing document formats and converting unstructured text into a form suitable for model input. When cleaning the data we preserve relevant clinical information, such as mentions of diabetes type and diagnosis year, while removing headers, and formatting artifacts that could interfere with automated extraction. Data cleaning is an important step in improving model performance and ensuring that the evaluation reflects the model's ability to extract clinically meaningful information from real world text.

patientnr	datum	diagrel	code	specialism	omschrijving
P005	1986-01-01	V	INT001	INT	Diabetes mellitus type X, sinds 1995 insuline pomp therapie
P005	1995-01-01	V	INT002	INT	Laparoscopische sterilisatie
P005	1999-01-01	V	INT003	INT	CTS rechts
P005	2001-01-01	V	INT004	INT	Buikklachten obv obstipatie
P005	2011-01-01	V	INT005	INT	Chronische primaire insomnia
P005	2012-01-01	V	INT006	INT	Subklinische hypothyreoidie
P005	2015-10-27	D	DBC001	INT	Diabetes mellitus chronisch pomptherapie @DBC
P005	2016-07-21	D	DBC002	ORT	Tendinitis van heup
P005	2017-08-02	D	DBC003	DER	Alopecia telogeen effluvium
P005	2019-06-12	D	DBC004	CHI	Fractuur van femurhals links
P005	2019-06-12	D	DBC005	CHI	Fractuur van femurhals rechts
P005	2020-07-03	D	DBC006	CHI	NULL

Table 3.5: Illustration of voorblad document before cleaning

patientnr	diagnostic_text
P005	1986-01-01 – Diabetes mellitus type X, sinds 1995 insuline pomp therapie
P005	1995-01-01 – Laparoscopische sterilisatie
P005	1999-01-01 – CTS rechts
P005	2001-01-01 – Buikklachten obv obstipatie
P005	2011-01-01 – Chronische primaire insomnia
P005	2012-01-01 – Subklinische hypothyreoidie
P005	2015-10-27 – Diabetes mellitus chronisch pomptherapie @DBC
P005	2016-07-21 – Tendinitis van heup
P005	2017-08-02 – Alopecia telogeen effluvium
P005	2019-06-12 – Fractuur van femurhals links
P005	2019-06-12 – Fractuur van femurhals rechts
P005	2020-07-03 – NULL

Table 3.6: Illustration of voorblad document after cleaning

3.6.1 Cleaning Voorblad

3.6.1.1 Explanation of Cleaning Steps

The original voorblad contained multiple structured fields, including the diagnosis relationship (`diagrel`), internal code (`code`), medical specialty (`specialism`), and a free text description. For the purpose of data cleaning, only the patient identifier and the textual diagnostic history are required. Therefore, the cleaning process involved the following transformations:

- The `PATIENTNR` column was kept as the unique identifier for linking records.
- The `DATUM` and `OMSCRIJVING` fields were merged into a single chronological diagnostic text field, improving readability and simplifying downstream processing.
- Unnecessary metadata columns (`diagrel`, `code`, `specialism`) were removed.
- All dates were standardized to the format `YYYY-MM-DD`, to ensure consistency for temporal analysis.

3.6.2 Cleaning Naslag

Naslag required the most cleaning due to extraction noise:

patientnr	text	date	time	type_of_text	text_code	specialism_filepart_text	docters_code	docters_name
P002	\rtf1\ansi Diabetes mellitus type 1; Inreda AP	2022-02-02	09:21	Reden van komst / Verwijzing	C0001	DIA	D1234	Dr. A. Example

Table 3.7: Anonymized Naslag record (before cleaning)

patientnr	clean_text
P002	2022-02-02 09:21 – Diabetes mellitus type 1; Inreda AP

Table 3.8: Naslag after cleaning: merged and cleaned diagnostic text

3.6.2.1 Explanation of Cleaning Steps

The raw Naslag records often contain embedded RTF control sequences, HTML encoding, or formatting metadata that have no semantic meaning for downstream processing. To prepare these records for text extraction and annotation, the following steps were performed:

- Elements such as `\rtf1`, `\ansi`, curly-braced RTF blocks, and escaped characters were stripped using pattern-based filtering.
- Residual formatting symbols (e.g., trailing backslashes, semistructured markers, extra whitespace) were removed via a series of regular expressions.
- The cleaned free text content was combined with the `date` and `time` fields to produce a chronological diagnostic entry.
- A single unified `clean_text` field was produced in the format:

```
YYYY-MM-DD HH:MM - cleaned diagnostic text
```

3.6.3 Cleaning Zorgdomein

patientnr	doc_type	doc_id	doc_date	section	text
P001	zorgdomein	D001	2017-10-09	Episodelijst	15-05-2016, chronische aandoening X behandeld met medicatie Y, A12.34. 02-02-1990, metabole aandoening Z, specialist, T56.78
P001	zorgdomein	D001	2017-10-09	Probleemlijst	10-09-2017, klacht A (bijv. pijn gewricht), L45. 05-09-2017, klacht B (bijv. spierpijn), L67. 20-08-2017, klacht C (bijv. huidirritatie), S34. 11-06-2016, klacht D (bijv. wond arm), S92. 03-06-2015, klacht E (bijv. mobiliteitsprobleem), L23. 15-05-2011, klacht F (bijv. slaapprobleem, medicatie X voorgeschreven), P07. 07-03-2005, algemeen journaal, A88.

Table 3.9: Example of extracted data from a Zorgdomein Report (before cleaning)

patient_number	doc_id	doc_date	section	content
11013175	1005027344	2017-10-09	Probleemlijst	21-08-2017, subklinische hypothyreoidie wv euthyrax, A91.06 01-10-1986, Insuline-afhankelijke diabetes type 1, internist, T90.01
11013175	1005027344	2017-10-09	Episodelijst	09-10-2017, pijn hand, L12 09-10-2017, pijn voet rechts, L17 28-09-2017, Haartuitval, S23 07-07-2016, wond onderbeen, S97 28-05-2015, heup klachten, L13 10-06-2011, slaapprobleem mag 1x per 2mnd 60 temazepam, P06 02-04-2005, Algemeen journaal, A99

Table 3.10: Example of extracted data from a Zorgdomein Report (after cleaning)

3.6.3.1 Explanation of Cleaning Steps

Zorgdomein reports were originally provided as multi page PDFs containing heterogeneous sections and inconsistent formatting. A custom extraction pipeline was developed to transform these PDFs into clean, structured data. The following steps were performed:

- Patient number, document identifier, and document date were extracted from PDF headers.
- Only the *Episodelijst* and *Probleemlijst* sections were extracted because these contain rich chronological clinical information, including diabetes related diagnoses.
- Each bulletstyle entry from the PDF was converted into a structured line in the `content` field.
- All entries were standardized to a consistent format:

  `DD-MM-YYYY, description, diagnostic_code`

- Instead of keeping all events in one long string, each event was preserved as a separate line within the `content` cell using `\newline`.

Chapter 4

Method

4.1 Proposed Solutions

Over the years, different approaches have been used to extract information from clinical text. Early systems such as rule based methods and regular expressions, which were effective for simple, well structured patterns but soon became less flexible when dealing with inconsistent medical notes. More recently, transformer based architectures have become the state of the art, and large language models (LLMs) has changed clinical information extraction by allowing flexible, context aware analysis without extensive manual engineering.

With this background, two solutions are proposed. The first involves fine tuning a Named Entity Recognition (NER) model using MedRoBERTa.nl in combination with a rule based approach for temporal expression normalization. The second approach is a prompt based extraction using local instruction tuned language models.

4.1.1 Solution 1: Named Entity Recognition with MedRoBERTa and Rule based Temporal Extraction

The first proposed solution will be built with Named Entity Recognition based on MedRoBERTa.nl, a transformer model pre trained on Dutch medical and clinical texts. Recent research has validated the effectiveness of MedRoBERTa.nl in Dutch clinical settings [9]. This model will be fine tuned to detect two key entities: `DIABETES_TYPE` and `DIAGNOSIS_YEAR`. The `DIABETES_TYPE` entity includes phrases such as type 1 diabetes or type 2 diabetes while the `DIAGNOSIS_YEAR` entity refers to temporal expressions that indicate when the diagnosis occurred, such as 01-02-2022.

Once the model identifies relevant entities in the text, regular expressions will be used to extract temporal informations which is the year of the diagnosed diabetes. This temporal extraction step will be done using regular expressions and logic to handle common patterns observed.

This approach is expected to have a greater interpretability, particularly in the temporal extraction step, which is important for clinical validation. However, its performance will heavily depend on the availability and quality of annotated training data, and it may struggle with implicit or vaguely mentions of diagnosis details.

4.1.1.1 medroberta.nl

MedRoBERTa.nl is a transformer based language model specifically pretrained on large scale Dutch medical and clinical corpora. It follows the RoBERTa architecture, which is an optimized variant of BERT that employs dynamic masking, larger batch sizes, and more extensive training iterations. Because MedRoBERTa.nl is exposed to domain specific Dutch medical terminology during pretraining, it is better equipped to recognize linguistic patterns, abbreviations, and clinical phrasing commonly found in electronic health records (EHRs).

Compared to generic Dutch language models, MedRoBERTa.nl has demonstrated superior performance on downstream clinical NLP tasks such as entity recognition, relation extraction, and document classification [9]. Its medical domain pretraining substantially reduces the need for large annotated corpora during fine tuning, which is beneficial in clinical settings where manually labelled data is limited. This makes it a suitable foundation for identifying diabetes-related information such as diagnosis type and diagnosis year.

4.1.1.2 Training the Model

To fine-tune MedRoBERTa.nl for the extraction of `DIABETES_TYPE` and `DIAGNOSIS_YEAR`, a token-level annotated dataset was created based on the manual annotation table (Table 3.4). Each annotated sentence was converted into a HuggingFace-compatible token classification format. The processed dataset contains the following fields:

- `input_ids`: Numerical token identifiers produced by the MedRoBERTa.nl tokenizer. Each token from the sentence is mapped to an integer representing its position in the model's vocabulary.
- `attention_mask`: A binary mask indicating which tokens should be attended to by the model. A value of 1 marks real tokens, while 0 corresponds to padding tokens.
- `labels`: Integer-encoded entity tags assigned to each token, aligned with the model's vocabulary and training scheme. Non-entity tokens are assigned the label `O`, while entity tokens are assigned labels such as `B-DIABETES_TYPE`, `I-DIABETES_TYPE`, `B-DIAGNOSIS_YEAR`, etc.
- `text`: The original clinical sentence before tokenization. This is kept for debugging and traceability.
- `metadata`: Additional document level metadata not used by the model during training, but used for evaluation or inspection. This includes the manually assigned diabetes type and diagnosis year for the sentence.

The entity labels follow the standard BIO tagging scheme. In this scheme, the prefix `B-` (Begin) indicates the first token of an entity span, while `I-` (Inside) marks subsequent tokens that are part of the same entity. For example, in the phrase "type 2 diabetes," the token "type" would receive the label `B-DIABETES_TYPE` and the token "2" would receive `I-DIABETES_TYPE`. This distinction helps the model recognize multi-token entities and correctly separate adjacent entities within a sentence.

An example instance of the NER-ready dataset is shown below:


```
{
   "input_ids": [0,21,17,21,17,19002,30,1107,15352,3399,4128,2099,300,35,2],
   "attention_mask": [1,1,1,1,1,1,1,1,1,1,1,1,1,1,1],
   "labels": [0,0,0,0,0,0,0,0,0,1,2,1,2,0,0],
   "text": "1-1-1991: Voorgeschiedenis 1991 diabetes mellitus type 1?",
   "metadata": {
     "diagnosis_label": "type 1",
     "date_label": "1991-01-01"
   }
}
```


4.1.1.2.1 Training Procedure The model is fine tuned using the HuggingFace `Trainer` framework, which handles optimization, batching, checkpointing, and evaluation. The annotated dataset is first converted into a HuggingFace `Dataset`, tokenized using the MedRoBERTa.nl tokenizer, and batched using a `DataCollatorForTokenClassification`. During training, the model learns to predict token level labels using a supervised cross entropy loss function. At each epoch, the model is evaluated on a validation split, and the best performing checkpoint is selected based on the F1-score.

To obtain a more reliable estimate of the model's generalization performance, a 5 fold cross validation procedure is used. Instead of relying on a single train–test split, the dataset is partitioned into five folds, and the model is trained and evaluated five times, each with a different validation fold. This reduces variance in performance estimation, mitigates the influence of small dataset bias, and ensures that all annotated data contributes to both training and evaluation at least once.

After completing cross-validation, a final model is trained on the entire annotated dataset, but still reserves a small portion as a validation set for early stopping and model selection. Importantly, no test set is used during this stage, the final performance reporting is based only on the cross validation results, to ensure that the model does not see its test data during training.

4.1.1.3 Prediction Result and Comparing to Manual Labelling

After training, the fine-tuned model is applied to the cleaned text chunks extracted per patient. For each document, the model predicts token-level labels and reconstructs entity spans corresponding to `DIABETES_TYPE` and `DIAGNOSIS_YEAR`. These predictions are then compared to the manually annotated ground-truth labels.

Discrepancies between predictions and manual labels are examined qualitatively to identify common error patterns, including vague temporal expressions, ambiguous phrasing, or cases where diagnosis information is implied rather than explicitly stated. These analyses help determine the reliability of the approach for real-world clinical deployment.

Flowchart

Figure 4.1 shows the workflow of Solution 1, from raw document collection to model prediction and evaluation.

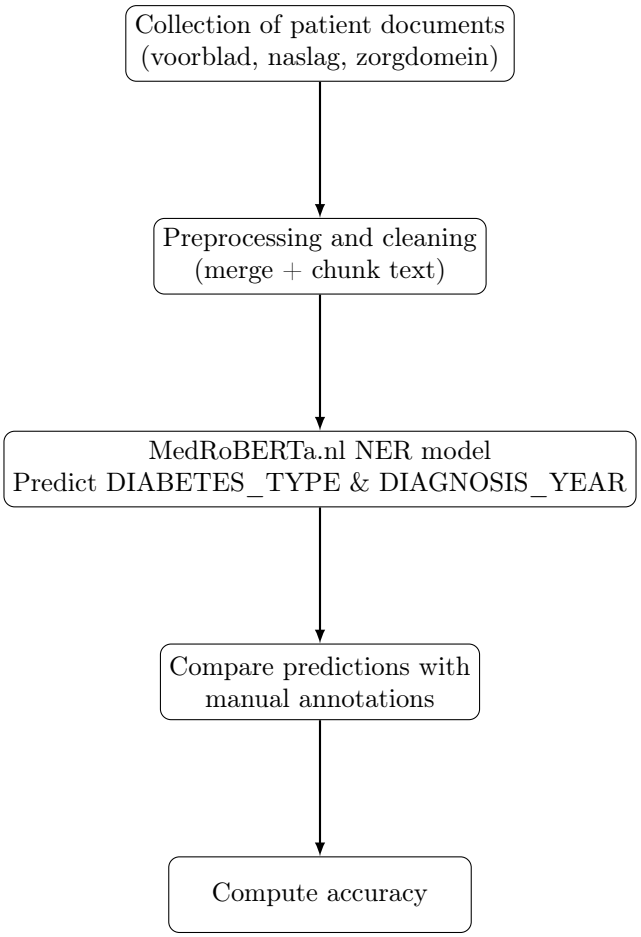


FIGURE 4.1: Flowchart of Solution 1: NER with MedRoBERTa.nl and rule-based temporal extraction.

4.1.2 Proposed Solution 2: Prompt based Extraction Using Local Language Models

The second solution takes a different approach by using prompt based extraction using instruction tuned local language models. Prompt based LLM extraction methods have shown promising results for clinical date and event extraction [13]. Instead of training a model for NER, this method plans to make specific prompts that instruct the language model to extract the diabetes type and year of diagnosis directly from a given input. For instance, given a sentence such as "The patient was diagnosed with type 2 diabetes in early 2011," a simple prompt like "Extract the diabetes type and diagnosis year" can be used to guide the model to return structured information such as "Type 2 diabetes, 2011."

This approach is simpler because it requires minimal training effort and avoids the need for large annotated datasets. It is also highly adaptable, because it can be used for quick iteration and refinement of prompts to suit new types of clinical documents or information needs.

However, the main challenges of this approach lie in ensuring consistency and reliability. Especially since the outputs of prompt based models are less deterministic and consistent compared to traditional NER systems, validating their performance across large scale clinical datasets can be more difficult. Also, the success of this method depends on the quality of the underlying language model and the effectiveness of prompt engineering. Furthermore, another risk in this approach is the potential for *hallucinations*, where the model generates plausible sounding but factually incorrect information, which can be especially problematic in the clinical domain. [6].

4.1.2.1 Ollama 3.1:8b

This solution uses the `Ollama 3.1:8b` model, a lightweight local language model optimized for running on consumer hardware. The model is instruction tuned, meaning it has been trained to follow user provided prompts and produce structured outputs. Running the model locally ensures full data privacy, which is essential in a clinical context. Additionally, local deployment avoids dependency on external APIs and provides consistent, low latency inference.

4.1.2.2 qwen2.5-7b

In addition to Ollama 3.1:8b, this solution also evaluates `qwen2.5-7b`, a language model that can be deployed locally using the same prompt based extraction pipeline. The main motivation for including `qwen2.5-7b` is to demonstrate the modularity of the proposed approach: the extraction logic is model agnostic, meaning that replacing the underlying LLM only requires changing the model identifier and a small number of configuration lines, while keeping the prompt template, parsing, and validation steps unchanged.

By testing multiple local LLMs under the same prompts and evaluation protocol, this thesis can compare robustness and consistency across models, and assess whether a stronger model yields more reliable structured outputs (diabetes type and year of diagnosis) without additional training. This also supports future extensibility, since newer local models can be integrated into the workflow with minimal engineering effort.

4.1.2.3 The Prompt

```
    Je bent een **medische data-extractie-assistent**. Analyseer de
    onderstaande Nederlandse medische documenten en haal informatie
    over **diabetesdiagnoses** eruit.

### CONTEXT EN HULP
- Jaartallen in de tekst zijn gemarkeerd als [YEAR]2023[/YEAR].
- Kies het **eerste logische jaartal** dat overeenkomt met het
   moment van **diagnose** van diabetes.
- Vermijd jaartallen die horen bij controle, behandeling of
   hulpmiddelen.

---

### VOORBEELDEN

Tekst:
"Diabetes mellitus type 1 sinds [YEAR]2002[/YEAR]. In [YEAR]2016[/
   YEAR] overstap naar insulinepomp."
Output:
{{
 "diabetes_type": "Type 1",
 "diagnosis_year": "2002",
 "year_source": "voorblad",
 "year_context": "Diabetes mellitus type 1 sinds 2002.",
 "type_source": "voorblad",
 "type_context": "Diabetes mellitus type 1"
```

```
}}

Tekst:
"DM2 vastgesteld in [YEAR]2010[/YEAR], laatste controle [YEAR]2015[/YEAR]."
Output:
{{
 "diabetes_type": "Type 2",
 "diagnosis_year": "2010",
 "year_source": "zorgdomein",
 "year_context": "DM2 vastgesteld in 2010",
 "type_source": "zorgdomein",
 "type_context": "DM2"
}}

Tekst:
"LADA diabetes, positieve anti-GAD65, sinds [YEAR]2013[/YEAR]."
Output:
{{
 "diabetes_type": "Type 1",
 "diagnosis_year": "2013",
 "year_source": "naslag",
 "year_context": "LADA diabetes, sinds 2013",
 "type_source": "naslag",
 "type_context": "LADA diabetes"
}}

---

### EXTRACTIEREGELS

1. **diabetes_type**     Altijd "Type 1" of "Type 2" (nooit null).
  Herken en normaliseer:
  - **Type 1**: "diabetes mellitus type 1", "dm type 1", "type 1 diabetes", "t90.01", "diabetes mellitus type i", "lada", "lada dm", "lada diabetes"
  - **Type 2**: "diabetes mellitus type 2", "dm type 2", "type 2 diabetes", "diabetes mellitus type ii"

2. **diagnosis_year**     Het **vroegste geldige jaartal (vier cijfers)** dat waarschijnlijk hoort bij de **eerste diagnose**.
  - Negeer jaartallen als "1900".
  - Kies het **eerste jaartal** dat vlak v   r of in de buurt van een diabetesvermelding staat.
  - Als meerdere jaren aanwezig zijn, kies het **eerste relevante jaartal** bij diagnose, niet bij behandeling of hulpmiddel.
  - Gebruik alleen het getal (bijv. "2012").

3. **year_source**     De bron (documentdeel) waarin het gekozen jaartal voorkomt: "voorblad", "zorgdomein" of "naslag".

4. **year_context**     De zin of tekstpassage waarin het gekozen jaartal voorkomt, of null indien onbekend.

5. **type_source**     De bron (documentdeel) waarin het type wordt genoemd.

6. **type_context**     De zin of tekstpassage waarin het diabetestype voorkomt.

---
```

```
72
73 ### SPECIALE INSTRUCTIES
74 - Gebruik ‘null‘ (zonder aanhalingstekens) voor ontbrekende velden,
       behalve bij ‘diabetes_type‘.
75 - Geef als output **uitsluitend geldig JSON** in exact het volgende
       formaat:
76
77 {{
78 "diabetes_type": "Type 2",
79 "diagnosis_year": "2012",
80 "year_source": "zorgdomein",
81 "year_context": "29-05-2012, Diabetes mellitus type 2 (keten)",
82 "type_source": "zorgdomein",
83 "type_context": "Diabetes mellitus type 2 (keten)"
84 }}
85
86 ---
87
88 ### PATIENTDOCUMENTEN
89 {combined_text}
90
91 ### GEEF TERUG (ALLEEN JSON):
```

The prompt used in this approach is designed as a structured instruction set that guides the language model to reliably extract diabetes related information from Dutch clinical documents. The goal is to ensure consistent outputs across heterogeneous sources (voorblad, naslag, zorgdomein) while minimizing hallucinations and enforcing strict output formatting. To achieve this, the prompt adopts a role based setup, explicit extraction rules, annotated examples, and a JSON schema that the model must follow.

First, the model is assigned the role of a “medische data-extractie assistent”, which frames the task in a clinically grounded context. The prompt provides additional guidance by highlighting that relevant years in the text are pre marked (e.g., 2023), which enable the model to focus attention on potential diagnosis dates. It also specifies that only the earliest year corresponding to the diabetes diagnosis should be selected, and that years associated with controls or treatment adjustments must be ignored.

To ensure clarity, the prompt includes several worked examples that demonstrate what the model should output for different phrasings, document structures, and diabetes terminology. These examples illustrate how to normalize diabetes types (e.g., mapping LADA to Type 1) and how to distinguish between mentions of diagnosis versus follow up. Following the examples, the prompt defines explicit extraction rules covering: normalization of diabetes type into either Type 1 or Type 2, selection of the earliest diagnosis year, retrieval of contextual segments and source metadata for both the year and the diabetes type.

Lastly, the prompt enforces strict JSON only output by specifying an exact schema. This is used in order to match the output of the first method, so a comparison can be made to determine accuracy against the manual labelling.

4.1.2.4 Prompt Size

In the Ollama-based implementation (`llama3.1:8b`), the prompt is constructed from three components: (i) a fixed instruction template containing the role definition, extraction rules, worked examples, and a JSON schema; (ii) a variable patient-specific text block (`combined_text`) that concatenates diabetes-relevant fragments from *voorblad*, *zorgdomein*, and *naslag*; and (iii) a short suffix that reiterates the requirement to return JSON only. The overall prompt size can therefore be expressed as:

$$T_{\text{total}} \approx T_{\text{instr}} + T_{\text{docs}} + T_{\text{suffix}},$$

where T_{instr} and T_{suffix} are constant across patients, and T_{docs} depends on the amount of diabetes-related text available for each patient.

Based on runtime token logging, the fixed instruction template contributes approximately **1K tokens** (about 1,008 tokens in our implementation) and the suffix contributes approximately **10–15 tokens**. Therefore, the *average* total prompt size is primarily determined by the average length

of `combined_text`. In typical cases, the prompt length is on the order of a few thousand tokens, i.e.,

$$T_{\text{total}} \approx 1{,}000 + T_{\text{docs}} \quad \text{tokens},$$

meaning that most of the token budget is available for clinical text rather than instructions. Although Llama 3.1 supports long context inference (up to 128K tokens as reported by Meta), our experiments used an input context of 8,192 tokens due to runtime configuration and resource constraints. With an input limit of **8,192 tokens**, this design leaves substantial room for longer patient histories, while still providing enough guidance (examples + rules + schema) to make sure consistent performance.

4.1.2.5 Workflow

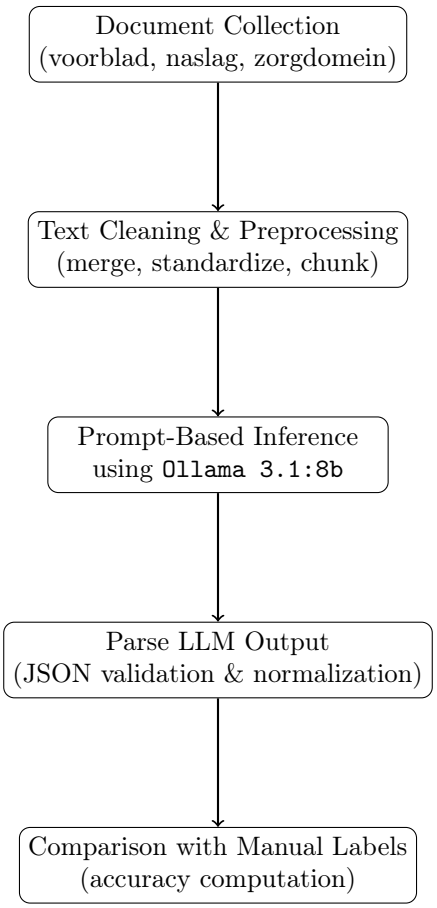


FIGURE 4.2: Workflow for Solution 2: Prompt-based Extraction using Local Language Models

The workflow of the second solution follows the same high level pipeline as the NER based approach. First, all available documents per patient, the voorblad, naslag, and zorgdomein entries are cleaned and merged.

Next, the documents went through a cleaning and preprocessing stage where formatting inconsistencies are removed, texts are standardized, and the content is segmented into manageable chunks. These chunks are then passed into the Ollama 3.1:8b model together with the carefully designed extraction prompt. The model produces structured JSON outputs that include the predicted diabetes type, diagnosis year, and the associated contextual metadata.

After inference, the outputs are parsed and validated to ensure they conform to the required JSON schema. This includes checking for missing fields, resolving inconsistencies, and normalizing values such as diabetes type labels. Finally, the extracted predictions are compared against manually annotated ground truth labels to assess the accuracy and reliability of the prompt based method. This evaluation step mirrors that of Solution 1, allowing for direct comparison between the NER-based and LLM-based extraction strategies.

4.1.2.6 Prediction Result and Comparing to Manual Labelling

The evaluation procedure mirrors that of Method 1. Model outputs are compared with the manually annotated diabetes type and diagnosis year for each patient.

4.1.3 Evaluation

The solutions are evaluated using both standard information extraction metrics and task specific evaluation metrics.

4.1.3.1 Diabetes Type Evaluation

For the extraction of diabetes type (categorical entity: *Type 1*, *Type 2*, or other/unspecified), performance is measured using accuracy. Accuracy is defined as the proportion of predictions that exactly match the manually annotated labels:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Accuracy is particularly appropriate for this task for several reasons:

- Each patient record has exactly one diabetes type, making this a single label, mutually exclusive classification problem. For such problems, a prediction is either fully correct or incorrect, and partial correctness (as captured by precision, recall, or F1) does not apply.
- The classes are relatively well represented in the dataset. Accuracy provides a clear and interpretable measure of overall model performance without the need to consider per-class weighting or complex averaging.
- Reporting accuracy aligns with the clinical relevance of the task: in practice, a diabetes type prediction is either correct or not, and the absolute proportion of correct predictions directly reflects the model's utility.

4.1.3.2 Diagnosis Year Evaluation

Evaluating the diagnosis year requires both strict and flexible measures:

- **Strict Precision, Recall, and F1-score:** A predicted year is counted as correct (*true positive*) only if it exactly matches the annotated year. Incorrect years count as *false positives*, while missed annotated years count as *false negatives*. This provides a strict evaluation comparable to other NLP information extraction tasks.
- **Error distance metrics:** Since strict evaluation treats being off by one year the same as being off by twenty years, additional metrics are reported:
  - **Mean Absolute Error (MAE):** the average absolute deviation between predicted and annotated years.
  - **Mean Squared Error (MSE):** the squared average deviation, which penalizes large errors more heavily.

4.1.4 Evaluation Procedure

Across all extraction methods, performance is evaluated at the patient level for three tasks:

- **Diabetes Type evaluation**: A patient is included if both the manual label and the model output contain a diabetes type.
- **Diagnosis Year evaluation**: A patient is included if both the manual label and the model output contain a diagnosis year.
- **Combined Type & Year evaluation**: A patient is included only if both fields (type and year) are present in both the manual labels and the model output.

The evaluation criteria are identical across all methods. However, the number of comparable patients may differ depending on whether a method outputs missing values for one of the entities. NER may fail to extract either the type or the year, resulting in fewer included patients for some evaluation tasks. Meanwhile, prompt based LLM methods typically output both fields together, which leads to identical patient counts across the three tasks.

Chapter 5

Results

This chapter presents the performance of the two proposed extraction methods: (1) a Named Entity Recognition (NER) model based on MedRoBERTa.nl, and (2) a prompt based local language model implemented using Ollama 3.1 (8B). Both methods were evaluated using the manually labelled dataset described earlier, which serves as the clinical ground truth for diabetes type and diagnosis year extraction.

The evaluation focuses on four metrics that directly reflect clinical usefulness: diabetes type accuracy, diabetes year accuracy, mean year difference, and combined type–year accuracy. These metrics were chosen because for diagnosis years, accuracy alone is insufficient because predicting "2014" instead of "2015" should not be penalized as heavily as predicting "1950." Therefore, the mean absolute difference between the predicted and annotated diagnosis year is also reported. Finally, combined accuracy serves as an overall measure of clinical reliability, requiring both the diabetes type and diagnosis year to be correct for a patient.

5.1 Method 1: Named Entity Recognition (NER)

As described in the evaluation procedure, the number of comparable patients may differ per task depending on which fields (type, year) are present in both the manual annotations and the model output. For the NER model, the resulting patient counts are shown in Table 5.1.

Table 5.1 summarizes the number of comparable patients used in each evaluation task.

Evaluation Task	**Comparable Patients**
Diabetes Type	498
Diagnosis Year	536
Combined Type & Year	497

TABLE 5.1: Number of comparable patients for each evaluation task.

In summary, the evaluation counts differ because each task requires a different subset of patients for which the necessary information is available in both the manual annotations and the model predictions. Missing information on either side leads to the exclusion of some patients, resulting in the different totals reported for each evaluation metric.

5.1.1 Result of the type and year accuracy

Table 5.2 presents the performance of the MedRoBERTa.nl NER model. The model achieved the highest overall performance among the evaluated methods.

Metric	**Score**
Diabetes Type Accuracy	88.8%
Diabetes Year Accuracy	83.6%
Mean Year Difference	1.16 years
Combined Type + Year Accuracy	75.9%

TABLE 5.2: Evaluation results for Method 1 (MedRoBERTa.nl NER).

The model correctly predicted the diabetes type for 442 out of 498 patients (88.8%). Year extraction also performed strongly, with an exact match accuracy of 83.6% and a low mean absolute error of 1.16 years. Most errors were minor (1–4 years), although a small number of cases exhibited larger deviations caused by placeholder dates (e.g., "1966-01-01") or ambiguous year mentions.

The combined accuracy of 75.9% indicates that for more than three quarters of patients, the model extracted both diabetes type and year correctly. This strong performance showcases the stability and determinism of the NER approach.

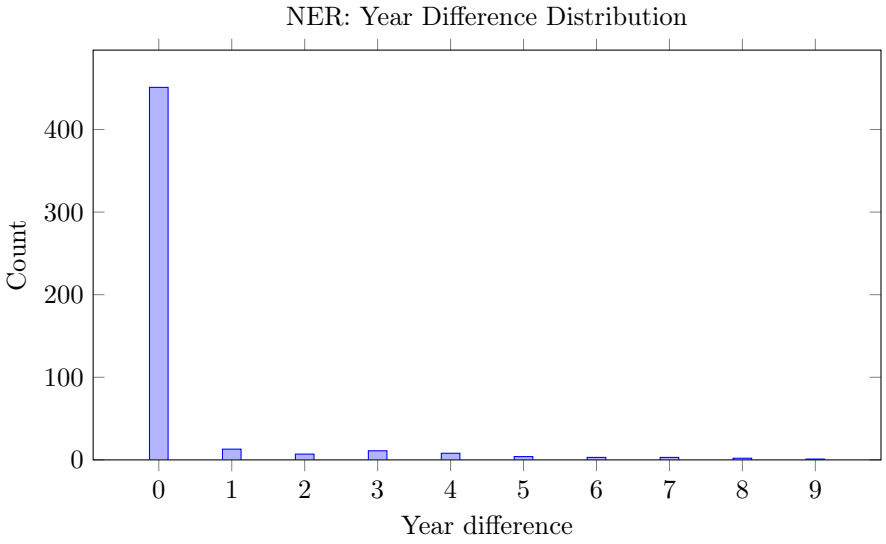


FIGURE 5.1

The NER model shows very strong performance on diagnosis year extraction, with 451 out of 536 patients having an exact match (year difference = 0). Only a small number of cases deviate by more than 4 years, indicating highly stable extraction of structured year information.

5.1.2 Interpretation of the Combined Type & Year Accuracy

The combined accuracy is not computed as the average of the Type accuracy and the Year accuracy. Instead, it reflects the proportion of patients for which the model correctly predicted *both* the diabetes type and the diagnosis year. Formally,

$$\text{Combined Accuracy} = \frac{\text{Number of patients with both type and year correct}}{\text{Number of patients comparable for both fields}}.$$

This means that a patient is counted as correct only when the predicted diabetes type *and* the predicted diagnosis year both match the manual annotation. If either the type or the year is incorrect, the entire combined prediction is considered incorrect.

Because joint correctness is a stricter requirement than correctness of the individual fields, the combined accuracy is always lower than the separate Type and Year accuracies.

This behavior can be understood probabilistically. If type correctness is viewed as an event T and year correctness as an event Y, then the combined accuracy corresponds to the probability of the intersection $P(T \cap Y)$. Under an independence assumption (used only for intuition), the expected joint accuracy would approximate $P(T) \cdot P(Y)$:

$$0.888 \times 0.836 \approx 0.747,$$

which is close to the observed combined accuracy of 0.759.

5.2 Method 2: Prompt-Based Extraction

5.2.1 Prompt-Based Extraction (LLM – All Documents)

Table 5.3 summarizes the number of comparable patients for each evaluation task.

Evaluation Task	Comparable Patients
Diabetes Type	576
Diagnosis Year	576
Combined Type & Year	576

TABLE 5.3: Comparable patient counts for the LLM (all documents) evaluation.

Following the standard evaluation criteria, the LLM produced complete outputs (type and year) for nearly all predicted patients. As a result, all three evaluation tasks use the same 576 comparable patients (Table 5.3).

Using all available clinical document types as input (Voorblad, Naslag, and Zorgdomein), the local instruction tuned LLM (Ollama 3.1, 8B) achieved the results shown in Table 5.4.

Metric	Score
Diabetes Type Accuracy	81.8%
Diabetes Year Accuracy	60.1%
Mean Year Difference	6.6 years
Combined Type + Year Accuracy	57.3%

TABLE 5.4: Evaluation results for Method 2 (LLM extraction using all document types).

The model reached a type accuracy of 81.8% across 576 detected patients, performing well for type 1 and type 2 cases.

Year extraction performance was substantially lower. The mean year difference increased sharply to 6.6 years, driven by a subset of large errors (up to 120 years). These extreme deviations were primarily caused by long documents containing conflicting or outdated information that misled the model.

Combined accuracy dropped to 57.3%, reflecting the sensitivity of the prompt-based approach to document length, noise, and implicit temporal references.

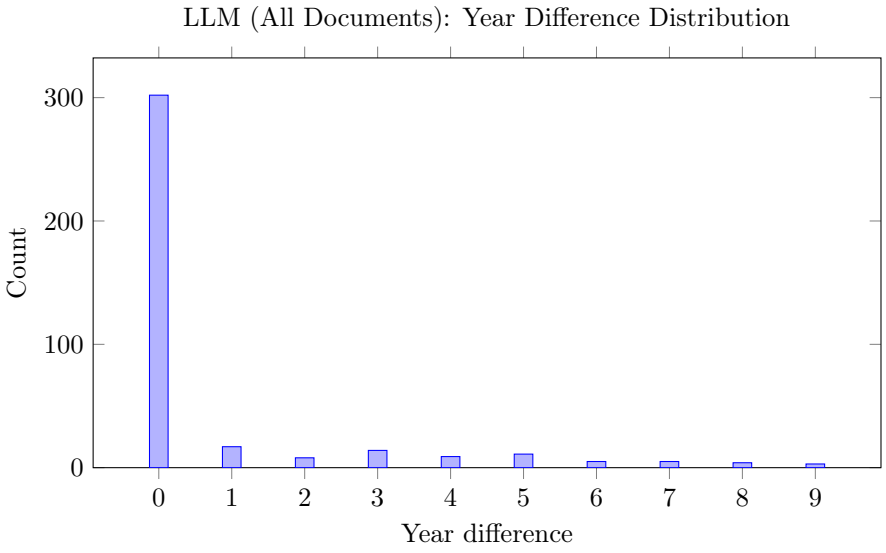


FIGURE 5.2

Using all documents, the LLM achieves only moderate year extraction accuracy. Although 346 cases have an exact match, the mean year difference increases substantially to 6.6 years. The long tail reflects the model being influenced by noisy or irrelevant year mentions scattered across multiple document types.

5.2.2 Method 2 (Variant): Prompt-Based Extraction Using Only Zorgdomein Documents

For the LLM evaluated using only Zorgdomein documents, the number of comparable patients is consistent across all evaluation tasks. Although the manual dataset contains 588 patients, the LLM produced predictions for only 182 patients, all of which overlap with the manual annotation.

Table 5.5 summarizes the number of comparable patients for each evaluation task.

Evaluation Task	Comparable Patients
Diabetes Type	182
Diagnosis Year	182
Combined Type & Year	182

Table 5.5: Comparable patient counts for the LLM (Zorgdomein only) evaluation.

The LLM produced type and year predictions for all 182 patients meeting the evaluation criteria, resulting in identical patient counts across all tasks (Table 5.5).

Metric	Score
Diabetes Type Accuracy	81.4%
Diabetes Year Accuracy	78.6%
Mean Year Difference	1.74 years
Combined Type + Year Accuracy	68.1%

Table 5.6: Evaluation results for Method 2 (LLM extraction using only Zorgdomein documents).

Restricting inputs to Zorgdomein significantly improved year extraction: accuracy increased to 78.6%, and the mean year difference decreased to 1.74 years comparable to the NER model. Most errors were again small (2–5 years), though several large deviations remained, largely due to cases where the referral letter mentioned only treatment changes rather than diagnosis timing.

Diabetes type accuracy decreased slightly to 81.4%, likely due to referral letters containing fewer explicit diagnostic statements compared to Voorblad or Naslag documents.

The combined accuracy of 68.1% shows that when the input is carefully controlled, the LLM can perform reasonably well, particularly for temporal extraction.

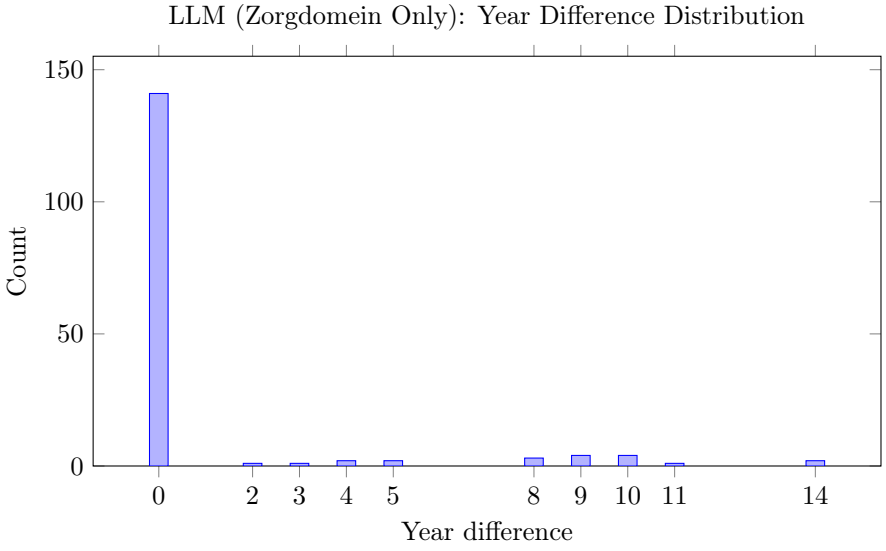


Figure 5.3

When restricting the input to Zorgdomein documents, the LLM performs much better: 143 of 182 patients yield an exact match and the mean year difference drops to 1.74 years. These structured referral letters appear to be more reliable sources for extracting diagnosis year than general medical notes.

Note on patient count. The Zorgdomein only evaluation contains far fewer patients (182) compared to the full dataset (500+). This is because not all patients in the EHR have Zorgdomein referral documents. As a result, the LLM can only evaluate the subset of patients for whom Zorgdomein documents actually exist.

5.2.3 Method 2 (Variant): Prompt-Based Extraction Using Only Voorblad Documents

Using only the Voorblad documents, the LLM produced predictions for 552 patients out of the 588 manually annotated cases. (Table 5.7).

Evaluation Task	Comparable Patients
Diabetes Type	552
Diagnosis Year	552
Combined Type & Year	552

TABLE 5.7: Comparable patient counts for the LLM (Voorblad only) evaluation.

Table 5.8 presents the performance results for Method 2 (Voorblad-only condition).

Metric	Score
Diabetes Type Accuracy	83.0%
Diabetes Year Accuracy	59.6%
Mean Year Difference	8.42 years
Combined Type + Year Accuracy	55.4%

TABLE 5.8: Evaluation results for Method 2 (LLM extraction using only Voorblad documents).

For diabetes type classification, the LLM correctly identified the type for 472 out of 552 patients, achieving an accuracy of 83.0%. This closely matches the performance observed when all document types were included.

Diagnosis year extraction remained the weakest aspect of the prompt based method. The model exactly matched the annotated diagnosis year in 304 of 476 cases (59.6%). The mean absolute year difference increased slightly to 8.42 years, with the largest error reaching 120 years. These outliers were typically associated with placeholder years (e.g., "1900"), incomplete temporal information, or the model selecting unrelated historical year mentions present in the Voorblad.

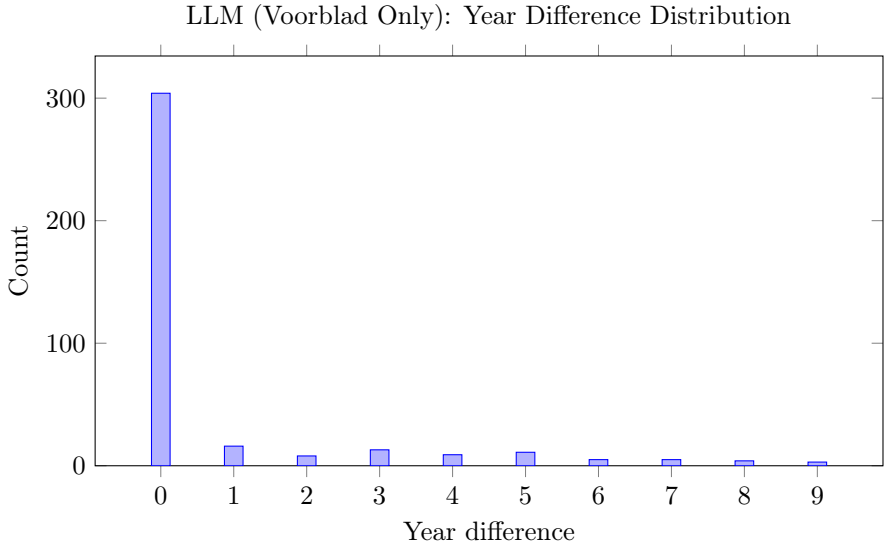


FIGURE 5.4: Year difference distribution for LLM extraction using only Voorblad documents.

The combined task, requiring both a correct diabetes type and an exact diagnosis year, resulted in an accuracy of 55.4% (272 of 476 patients). This highlights the compounded difficulty of simultaneously extracting categorical and temporal information from relatively short but often noisy front-page summaries.

Compared with the Zorgdomein only evaluation, Voorblad inputs led to improved performance for diabetes type extraction but substantially worse temporal accuracy. This suggests that although Voorblad summaries contain strong diagnostic cues, their dense and ambiguous year mentions make them less reliable for reconstructing diagnosis timelines.

5.2.4 Method 2 (Variant): Prompt-Based Extraction Using Qwen 2.5-7b

This variant of Method 2 replaces the original LLM with Qwen 2.5–7B while keeping the same prompt-based extraction setup. Using the full set of available documents, Qwen produced comparable predictions for 576 patients for diabetes type evaluation.

Evaluation Task	Comparable Patients
Diabetes Type	576
Diagnosis Year	576
Combined Type & Year	576

TABLE 5.9: Comparable patient counts for the Qwen 2.5–7B evaluation (Method 2 variant).

Table 5.10 presents the performance results for the Qwen 2.5–7B prompt-based extraction variant.

Metric	Score
Diabetes Type Accuracy	86.3%
Diagnosis Year Accuracy	74.5%
Mean Year Difference	1.97 years
Combined Type + Year Accuracy	71.7%

TABLE 5.10: Evaluation results for Method 2 (prompt-based extraction) using Qwen 2.5–7B.

For diabetes type classification, Qwen correctly identified the type for 497 out of 576 patients, achieving an accuracy of 86.3%.

Diagnosis year extraction improved substantially compared to earlier prompt-based variants. The model exactly matched the annotated diagnosis year in 429 of 576 cases (74.5%), with a mean absolute year difference of 1.97 years. This indicates that, when the model did not find an exact year match, its predictions were typically close to the annotated year rather than drifting toward unrelated historical mentions.

For the combined task, requiring both a correct diabetes type and an exact diagnosis year, Qwen achieved an accuracy of 71.7% (413 of 576 patients). Overall, these results suggest that Qwen 2.5–7B provides stronger temporal extraction performance while maintaining high diabetes type accuracy, making it a more reliable prompt-based baseline for structured extraction from clinical narratives.

5.3 Comparison Between NER and LLM Approaches

To better understand the differences between the NER and LLM extraction methods, we compare performance across multiple evaluation metrics: diabetes type accuracy, diagnosis year accuracy, mean year difference, and combined type–year accuracy. Figures 5.5 and 5.6 provide a visual summary.

5.3.1 Accuracy Comparison

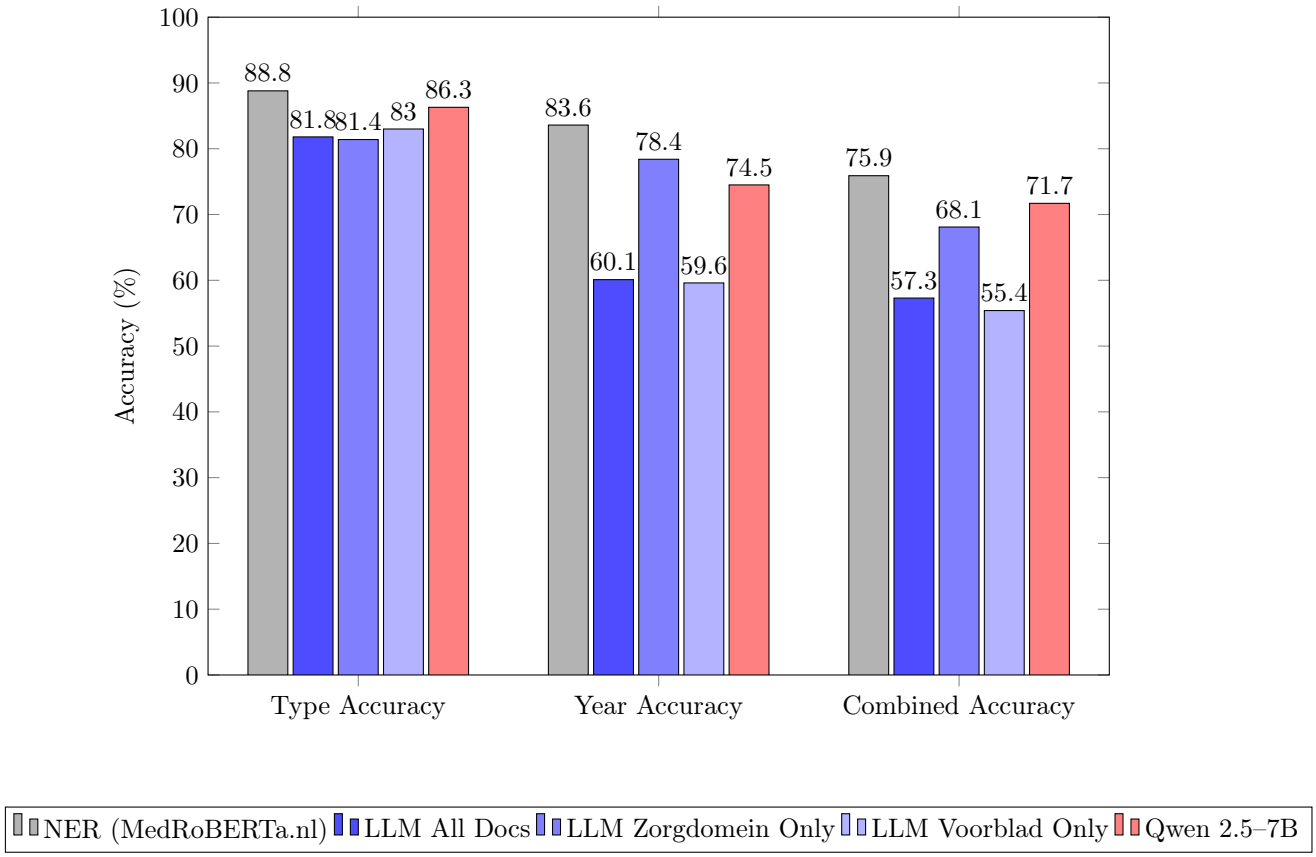


FIGURE 5.5: Comparison of diabetes type, diagnosis year, and combined type–year accuracy between NER and LLM methods.

The grouped bar chart shows that the NER model consistently outperforms all LLM variants in both type and combined accuracy. The LLM shows improved diagnosis year extraction when restricted to Zorgdomein documents, highlighting the importance of selecting informative document types for LLM-based extraction.

5.3.2 Mean Year Difference Comparison

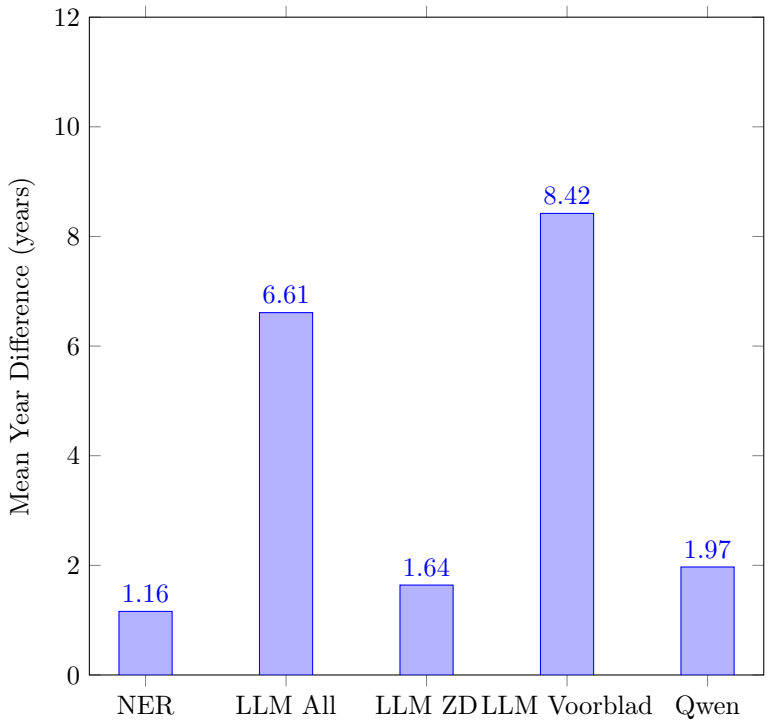


FIGURE 5.6: Mean absolute difference between predicted and annotated diagnosis years for NER and LLM approaches. Smaller values indicate better temporal accuracy.

The mean year difference highlights the temporal extraction limitations of the LLM when using noisy or short document inputs. Restricting the LLM to Zorgdomein documents reduces the mean error to near NER levels (1.64 years vs 1.16 years). In contrast, LLM performance remains poor when using all documents or Voorblad only, with extreme outliers driving the mean difference above 8 years.

These visualizations show several key points:

- **NER stability:** The NER model achieves the highest type and combined accuracy while maintaining low temporal error, showing consistent and reliable extraction performance.
- **LLM sensitivity:** LLM performance varies strongly with input selection. Zorgdomein only inputs improve year extraction substantially, but type accuracy is slightly lower, illustrating a trade-off between document scope and extraction quality.
- **Impact of document type:** Voorblad documents alone lead to poor year accuracy and high mean year difference, which suggests that this document may contain administrative dates or placeholder values that mislead LLM predictions.

Overall, these results support the conclusion that while LLMs are flexible and capable of reasoning over unstructured text, carefully curated document inputs are required to achieve clinically meaningful extraction performance. NER approaches, by contrast, offer higher determinism and reliability, making them preferable for large-scale deployment in EHR systems.

5.3.3 Overall disagreement distribution

Table 5.11 summarises the outcomes. The dominant pattern is *LLM wrong, NER correct* (101/105; 96.2%). Only 4 cases (3.8%) show the reverse pattern (*LLM correct, NER wrong*). Note that this dataset was constructed to contain disagreement cases, which explains why there are no both correct cases in this subset and should not be interpreted as the systems never agreeing in the broader corpus.

Table 5.12 shows representative disagreement examples. The "Context" column illustrates where the incorrect year was sourced from.

Table 5.11: Disagreement outcomes between LLM and NER for year-of-diagnosis extraction.

Outcome category	Count	Percent
LLM wrong, NER correct	101	96.2%
LLM correct, NER wrong	4	3.8%

Table 5.12: Representative disagreement examples (context truncated for readability).

GT	LLM	NER	Wrong method	Context where wrong year was taken
1998	2012	1998	LLM	`29-05-2012, Diabetes mellitus type 2 (keten)`
1998	2015	1998	LLM	`2015-11-15 diabetes mellitus type 1`
2010	2016	2010	LLM	`2016-02-18 diabetes mellitus type 1`
2008	2008	2007	NER	`2007-01-01 ... diabetes mellitus type 1, sinds ...`
2014	2014	2015	NER	`2015-02-17 DM april 2014`
2019	2019	2020	NER	`15-01-2020, Diabetes mellitus type 1 (2019)`

5.3.4 When the LLM is wrong but NER is right

In the disagreement set, the most common LLM error is that it selects a distractor date instead of the true year of diagnosis. Clinical notes often contain multiple time expressions: for example the consultation date, admission date, procedure dates, or dates of follow up visits. These dates are frequently written in a visually prominent format (e.g., `YYYY-MM-DD`). When such a date appears close to a mention of diabetes, the LLM may incorrectly interpret it as the diagnosis year. In practice, this means the LLM can output the encounter year rather than the diagnosis year.

Finally, a subset of LLM errors contains an implausible value such as **1900**. This is unlikely to be a clinically meaningful diagnosis year and more consistent with a fallback or parsing artefact (e.g., a default year returned when extraction fails). These cases should therefore be interpreted separately from genuine temporal confusion between multiple years in the note.

5.3.5 When NER is wrong but the LLM is right

NER failures are rare in this set, but they follow predictable patterns. First, in notes containing multiple competing time expressions, NER may refer onto the closest or most date like token (e.g., a full date `YYYY-MM-DD`) rather than the diagnosis year. Second, parenthetical years or "since" statements can create ambiguous local contexts where a rule- or span-based method associates the year with the wrong anchor. In these situations, the LLM can outperform NER by leveraging broader semantics (e.g., interpreting "*DM april 2014*" as an explicit diagnosis time statement).

Chapter 6

Discussion

6.1 Performance of the NER Approach

The MedRoBERTa.nl based NER model shows strong and stable performance, particularly for extracting diabetes type. With an accuracy of 88.8%, the model reliably distinguishes between type 1 and type 2 diabetes. The NER model's architecture, which uses span level predictions grounded directly in the text, ensures that extracted labels remain interpretable, which is an important property for clinical applications where traceability is essential.

Year extraction was similarly robust. The model achieved 83.6% exact match accuracy with a mean absolute difference of only 1.16 years. Importantly, most deviations between prediction and ground truth were minor (1–4 years), and large outliers were rare. This means that the model is effective at identifying both explicit date mentions and contextual clues that anchor diagnosis timing. Because the NER model evaluates each token locally rather than synthesizing long range reasoning, it is less prone to hallucinating or misinterpreting temporal references. This stability makes the NER approach highly predictable across diverse documentation styles.

6.2 Performance of the LLM Approach

The prompt-based LLM approach shows high variability depending on the input document type. For LLM – Ollama, when all documents were provided as input, the model achieved a reasonable diabetes type accuracy of 81.8%, but performance deteriorated significantly for diagnosis year extraction. The mean year difference of 6.61 years reflects a pattern of severe outliers, with some errors exceeding 50–120 years. These mistakes typically occurred in longer documents containing multiple date fields, outdated entries, or conflicting temporal information. The model sometimes selected the earliest or most prominent date rather than the clinically relevant one.

Restricting the input to Zorgdomein-only documents for LLM – Ollama substantially improved temporal extraction, achieving 78.6% year accuracy and reducing the mean difference to 1.74 years. This suggests that shorter, more task-focused clinical summaries are better suited to LLM reasoning. Zorgdomein documents often contain clearer references to diagnosis history or treatment trajectories, reducing ambiguity compared to multi-document clinical notes. However, type accuracy decreased slightly to 81.4%.

Using Voorblad-only documents for LLM – Ollama showed similar diabetes type extraction performance to the all-documents variant, with type accuracy at 83.0%. However, diagnosis year extraction again suffered from errors, with only 59.6% exact year matches and a mean year difference of 8.42 years. Combined type and year accuracy was 55.4%, reflecting the difficulty of extracting both categorical and temporal information from short, noisy summaries.

In addition to the Ollama-based variants, LLM – Qwen (Qwen 2.5–7B) was evaluated using a prompt-based extraction setup. As shown in Table 5.10, Qwen achieved a diabetes type accuracy of 86.3% and a diagnosis year accuracy of 74.5%, with a mean year difference of 1.97 years. The combined type + year accuracy was 71.7%, indicating that Qwen can extract both fields more reliably than the weaker Ollama variants, particularly for temporal information, while still remaining below the NER approach on the combined metric.

Overall, the LLM results indicate that while these models can infer implicit information and generalize without supervised training, their performance is less when faced with clinical documentation with multiple date mentioned. Document selection has a strong impact on robustness,

especially for year extraction. In addition, the less explainable nature of generative AI is also a challenge, as repeated runs may produce slightly different interpretations.

6.3 Error Analysis

This section analyses disagreement cases between the large language model (LLM) and the named entity recognition (NER) approach for extracting the *year of diabetes diagnosis*.

6.3.1 Strengths and weaknesses of each method

Overall, the LLM and NER approaches have complementary strengths for extracting the year of diabetes diagnosis from Dutch clinical notes. The main strength of the **LLM** is that it can handle many writing styles (free text, abbreviations, or unusual phrasing). This helps especially when the diagnosis year is not written in a neat, fixed pattern (e.g., when it is implied in a narrative). However, the LLM is also more likely to be distracted by other dates in the note. For example, it may pick the consultation date (often written as `YYYY-MM-DD`) simply because it is very visible and close to the word "diabetes", even though it is not the diagnosis date. In some cases the LLM can also return an implausible year such as **1900**, which likely indicates a fallback or processing artefact rather than a real year in the text.

The main strength of **NER** is that it is more consistent and easier to trace back to the original text span: it often performs well when the diagnosis year is explicitly stated near clear cues such as *sinds* (since), *diagnose*, or shorthand like *DM* followed by a time expression (e.g., "DM april 2014"). This makes NER less sensitive to prominent encounter dates and therefore more robust in many structured or semi structured note fragments. Its weakness is that it has less semantic understanding: when multiple dates appear close together (timelines, parentheses, lists), NER can select the wrong date because it looks more like a standard date or happens to be nearby, even if it refers to a follow up event instead of the diagnosis.

6.3.2 Why can NER outperform an LLM in our setting?

Although LLMs are often assumed to be "stronger" than traditional NLP models, this does not imply that they will outperform a supervised NER pipeline on a narrow, strictly scored clinical extraction task. Our LLM setup (Llama 3.1 8B) is effectively *few-shot* (a single instruction prompt with a few in prompt labeled examples), whereas the NER model is directly optimized for span/label prediction under supervised learning, creating a stronger bias that aligns with our evaluation objective. This matters because our task requires consistent extraction of a small, fixed set of fields (diabetes type and exactly one diagnosis year), where deterministic structured prediction is often more reliable than generative text completion. The largest gap in our results occurs for year extraction (LLM year accuracy 59.6% vs. NER 83.6%, and mean year difference 6.61 vs. 1.16 years), which is consistent with known limitations of LLMs in temporal reasoning: selecting the correct temporal anchor among many dates (visit dates, lab dates, medication start dates) are more difficult, and temporal reasoning has been explicitly identified as a challenging area for LLMs requiring dedicated mitigation strategies [12]. Finally, LLM outputs can also risks hallucinations or overconfident guesses when evidence is weak, a risk that has been highlighted in clinical evaluations of LLM behavior [2]. In combination, these factors explain why a domain specific supervised NER pipeline can outperform a prompted 8B LLM for this specific clinical information extraction setting, especially on the temporally grounded "year of diagnosis" field.

6.3.3 How could the LLM baseline be improved (given our few-shot prompt)?

Because our LLM (Llama 3.1 8B via Ollama) already uses a few-shot prompt with explicit extraction rules and JSON examples, the remaining performance gap, especially for diagnosis year extraction is likely driven less by "not knowing the task" and more by limitations in temporal selection, context management, and output reliability. Several improvements are therefore primarily systems and prompt engineering oriented. First, improve chunking: rather than feeding the entire patient document bundle at once, split documents into smaller chunks and retrieve only the most relevant sentences (e.g., those containing "DM", "diabetes", "LADA", "vastgesteld", "sinds", and marked years). Retrieval augmented generation (RAG) reduces distraction from irrelevant dates (labs, visits, medication changes) and improves evidence focused extraction [8]. Second, consider

domain adaptation or supervised fine tuning of the LLM on Dutch clinical notes or labeled extraction examples; domain adaptive pretraining and task fine tuning have shown to improve reliability on specialized domains [5, 10].

Overall, these interventions provide concrete ways to improve the LLM baseline and should particularly reduce year selection errors.

6.4 Limitations

Both methods shows important limitations. The NER model's dependence on annotated training data and specific pipeline makes it less general. The annotated patient records used in this study, while sufficient for initial evaluation, may not represent the full diversity of Dutch EHR writing styles across hospitals, or even in the future. Additional annotation or domain adaptation may be required for deployment in new clinical environments.

The LLM method faces different limitations: less explainability, prompt sensitivity, and higher computational cost (approximately 42 seconds per patient, using CPU only). Its performance variability across document types reveals a strong dependency on input quality. Moreover, the LLM occasionally generated plausible but incorrect years, which also highlight a risk of hallucination. In clinical settings, such errors could have significant implications if not properly validated.

Additionally, placeholder dates such as "1900-01-01" or "1966-01-01" created artifacts that particularly misled the LLM.

6.5 Implementability in Clinical Settings

From a deployment perspective, the NER model is the more practical choice for integration into electronic health record (EHR) pipelines. Its deterministic outputs, low inference cost, and interpretability align well with clinical system requirements. The model can be executed in real time and scaled to hundreds of thousands of records with minimal hardware.

The LLM, however, may be more useful in scenarios requiring contextual reasoning or richer summarization beyond structured extraction. Its ability to infer diagnosis type or timing from indirect descriptions is valuable, but operational constraints such as runtime cost, variability, and explainability, limit its suitability for real time or large scale extraction tasks. To use an LLM reliably in clinical workflows, additional attention such as input filtering and validation layers would be required.

6.6 Comparison to Existing Literature

Prior research in clinical NLP consistently shows that supervised models, including BERT based Named Entity Recognition systems, achieve strong and stable performance for structured clinical information extraction tasks [7, 11, 3, 9]. Recent work has also explored the use of Large Language Models for clinical information extraction, which demonstrate their flexibility in low resource or few shot settings, while also mentioning sensitivity to prompt formulation and contextual input [1, 13].

Studies on temporal information extraction from electronic health records report persistent challenges related to ambiguous and conflicting date mentions across clinical notes [4], which is consistent with the large year difference errors observed in this study. Our findings align with these observations and further emphasize the importance of selecting appropriate document types when applying instruction tuned LLMs in multilingual clinical settings.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

This thesis investigated the use of natural language processing (NLP) techniques to extract the type and year of diabetes diagnosis from unstructured Dutch electronic health record (EHR) texts. The study compared two approaches: (1) a supervised Named Entity Recognition (NER) pipeline based on MedRoBERTa.nl and (2) prompt-based extraction using locally deployable large language models (LLMs).

RQ1: To what extent can NLP be used to reliably extract the type and year of diabetes diagnosis from unstructured Dutch EHR texts, while resolving conflicting or misleading information?

The results show that NLP can extract both diabetes type and diagnosis year with clinically meaningful accuracy, but reliability depends strongly on the method and the level of noise in the input. The NER approach achieved the most stable overall performance, reaching 88.8% diabetes type accuracy, 83.6% exact year accuracy, a mean absolute year error of 1.16 years, and 75.9% combined type and year accuracy. These results indicate that for more than three quarters of patients, both fields can be extracted correctly using the NER pipeline.

In contrast, prompt-based LLM extraction was more sensitive to misleading temporal information, particularly when long notes contained multiple dates such as encounter dates, procedure dates, or placeholder dates (for example 1900-01-01 or 1966-01-01). These cases led to large outliers and reduced reliability.

RQ1.1: Which is more effective for extracting diabetes diagnosis year and type from Dutch EHRs: rule-based NER approaches or prompt-based LLM approaches?

Overall, the MedRoBERTa.nl-based NER approach outperformed the prompt-based LLM baselines on both diabetes type and combined extraction reliability. While the LLM setup requires less supervised training data and is flexible, its performance was brittle under noisy inputs. For example, when the LLM (Ollama 3.1 8B) used all document types, it achieved 81.8% type accuracy but only 60.1% year accuracy, with a much higher mean year error of 6.6 years, resulting in 57.3% combined accuracy. A stronger prompt-based variant using Qwen 2.5 7B improved year extraction to 74.5% and achieved 71.7% combined accuracy, but still remained below the NER method on the combined metric.

RQ1.2: Which documents provide the most accurate and complete information for extracting the diagnosis type and year?

Document selection strongly affected prompt-based extraction. For the LLM (Ollama 3.1 8B), restricting input to Zorgdomein-only documents substantially improved temporal performance to 78.6% year accuracy and reduced the mean year error to 1.74 years, indicating that shorter, more structured referral letters are less prone to misleading date selection.

However, for completeness and manual ground truth, both Voorblad and Zorgdomein are important sources.

In conclusion, this thesis demonstrates that automated extraction of diabetes type and diagnosis year from Dutch clinical narratives is feasible. For large scale clinical deployment where determinism, interpretability, and stability are required, the NER approach is the most practical solution, while LLM-based extraction may be most suitable when combined with input filtering and stronger validation mechanisms.

7.2 Future Work

While this study provides a foundation for extracting diabetes-related information from unstructured Dutch EHR text, several directions can improve robustness and deployability:

- Retrieval and input filtering before LLM extraction (RAG): Because LLM errors were often caused by distractor dates in long documents, future systems should retrieve only the most relevant sentences (for example containing DM, diabetes, LADA, vastgesteld, sinds, plus nearby years) before extraction. This reduces attention dilution and improves evidence-focused temporal selection.
- Chunking and aggregation across documents: Instead of extracting once from a merged document bundle, extraction could be performed per document or per section, followed by aggregation rules such as majority voting for type or selecting the diagnosis year closest to explicit diagnosis phrasing. This can reduce the impact of single noisy notes and improve consistency.

Bibliography

[1] Monica Agrawal et al. "Large Language Models are Few-Shot Clinical Information Extractors". In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2022. URL: https://arxiv.org/abs/2205.12689.

[2] Elham Asgari et al. "A framework to assess clinical safety and hallucination rates of LLMs for medical text summarisation". In: *npj Digital Medicine* (2025).

[3] M. C. Durango, E. A. Torres-Silva, and A. Orozco-Duque. "Named Entity Recognition in Electronic Health Records: A Methodological Review". In: *Healthcare Informatics Research* 29.4 (Oct. 2023), pp. 286–300. DOI: 10.4258/hir.2023.29.4.286.

[4] Sholle Fu et al. "Extracting and classifying diagnosis dates from clinical notes: a case study". In: *BMC Medical Informatics and Decision Making* 20.1 (2020), p. 189. DOI: 10.1186/s12911-020-01211-y. URL: https://pubmed.ncbi.nlm.nih.gov/32949781/.

[5] Suchin Gururangan et al. "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks". In: *ACL*. 2020.

[6] Ziwei Ji et al. "Survey of hallucination in natural language generation". In: *ACM Computing Surveys* 55.12 (2023), pp. 1–38.

[7] Kevin Kreimeyer et al. "Natural language processing systems for capturing and standardizing unstructured clinical information: A systematic review". In: *Journal of Biomedical Informatics* 73 (2017), pp. 14–29.

[8] Patrick Lewis et al. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks". In: *NeurIPS*. 2020.

[9] Hielke Muizelaar et al. "Extracting patient lifestyle characteristics from Dutch clinical text with BERT models". In: *BMC Medical Informatics and Decision Making* 24.151 (2024). DOI: 10.1186/s12911-024-02557-5. URL: https://doi.org/10.1186/s12911-024-02557-5.

[10] Stella Verkijk and Piek Vossen. "MedRoBERTa.nl: A Language Model for Dutch Electronic Health Records". In: *Computational Linguistics in the Netherlands Journal* (2021).

[11] Yanshan Wang et al. "Clinical information extraction applications: A literature review". In: *Journal of Biomedical Informatics* 77 (2018), pp. 34–49.

[12] Siheng Xiong et al. "Large Language Models Can Learn Temporal Reasoning". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*. 2024.

[13] Hao Zhang et al. "Evaluating Large Language Models in Extracting Cognitive Exam Dates and Scores". In: *medRxiv* (2024). Preprint. DOI: 10.1101/2023.07.10.23292373. URL: https://doi.org/10.1101/2023.07.10.23292373.