Ad Hoc Information Extraction for Clinical Data Warehouses

Georg Dietrich[1]; Jonathan Krebs[1]; Georg Fette[1,2]; Maximilian Ertl[3]; Mathias Kaspar[2]; Stefan Störk[2]; Frank Puppe[1]

[1]Computer Science, University of Wuerzburg, Wuerzburg, Germany;
[2]Comprehensive Heart Failure Center (CHFC), University Hospital of Wuerzburg, Wuerzburg, Germany;
[3]Service Center Medical Informatics, University Hospital of Wuerzburg, Wuerzburg, Germany

Keywords
Information extraction, information retrieval, Data Warehouse, clinical trials, negation detection, natural language processing

Summary

Background: Clinical Data Warehouses (CDW) reuse Electronic health records (EHR) to make their data retrievable for research purposes or patient recruitment for clinical trials. However, much information are hidden in unstructured data like discharge letters. They can be preprocessed and converted to structured data via information extraction (IE), which is unfortunately a laborious task and therefore usually not available for most of the text data in CDW.

Objectives: The goal of our work is to provide an ad hoc IE service that allows users to query text data ad hoc in a manner similar to querying structured data in a CDW. While search engines just return text snippets, our systems also returns frequencies (e.g. how many patients exist with "heart failure" including textual synonyms or how many patients have an LVEF < 45) based on the content of discharge letters or textual reports for special investigations like heart echo. Three subtasks are addressed: (1) To recognize and to exclude negations and their scopes, (2) to extract concepts, i.e. Boolean values and (3) to extract numerical values.

Methods: We implemented an extended version of the NegEx-algorithm for German texts that detects negations and determines their scope. Furthermore, our document oriented CDW PaDaWaN was extended with query functions, e.g. context sensitive queries and regex queries, and an extraction mode for computing the frequencies for Boolean and numerical values.

Results: Evaluations in chest X-ray reports and in discharge letters showed high F1-scores for the three subtasks: Detection of negated concepts in chest X-ray reports with an F1-score of 0.99 and in discharge letters with 0.97; of Boolean values in chest X-ray reports about 0.99, and of numerical values in chest X-ray reports and discharge letters also around 0.99 with the exception of the concept age.

Discussion: The advantages of an ad hoc IE over a standard IE are the low development effort (just entering the concept with its variants), the promptness of the results and the adaptability by the user to his or her particular question. Disadvantage are usually lower accuracy and confidence.

This ad hoc information extraction approach is novel and exceeds existing systems: Roogle [1] extracts predefined concepts from texts at preprocessing and makes them retrievable at runtime. Dr. Warehouse [2] applies negation detection and indexes the produced subtexts which include affirmed findings. Our approach combines negation detection and the extraction of concepts. But the extraction does not take place during preprocessing, but at runtime. That provides an ad hoc, dynamic, interactive and adjustable information extraction of random concepts and even their values on the fly at runtime.

Conclusions: We developed an ad hoc information extraction query feature for Boolean and numerical values within a CDW with high recall and precision based on a pipeline that detects and removes negations and their scope in clinical texts.

Correspondence to:
Georg Dietrich
University of Wuerzburg
Computer Science
Am Hubland
97070 Wuerzburg
Germany
E-mail: dietrich@informatik.uni-wuerzburg.de

Methods Inf Med 2018;57(Open 1): e22–e29
https://doi.org/10.3414/ME17-02-0010
received: July 28, 2017
accepted: February 10, 2018
Funding
This work was supported by the Comprehensive Heart Failure Center Würzburg (BMBF grants: #01EO1004 and #01EO1504).

1. Introduction

Common use cases for a CDW are to query frequencies of patients with certain inclusion and exclusion criteria, e.g. for assessing whether there are enough patients for a clinical trial. If a major part of the required data is not available as structured data but only included in textual reports, such assessments are quite time-consuming by manually checking many text documents. The standard method would be to preprocess the textual data within the ETL[1] process transferring data from the EHR into the CDW with information extraction methods. Various approaches to extract structured information from unstructured texts exist (e.g. for German texts [3, 4, 5]), but they require computational heavy preprocessing in the integration step and cannot be applied at query time dynamically. Furthermore much time has to be spend for engineering and building ontologies.

1 Extract, Transform, Load

1.1 CDWs and Their Extraction Features

Another approach is to retrieve the information dynamically at runtime. However, most CDWs does not support textual queries very well. That revealed a research in literature and websites of CDWs, their extensions, patient recruitment and clinical systems and research data bases: ArchiMed [6], BigQ [7], DW4TR [8], EHR4CR [9], Harvest [10], i2b2 [11], OpenMRS [12], REDCap [13], STRIDE [14], tranSMART [15], Vanderbilt [16], x4T [17], Roogel [1], Dr. Warehouse [2].

Most of them do not describe how textual data can be queried. A few systems provide information on how the data is stored. This allows conclusions on the possible text related query features. The CDW Harvest uses the data abstraction layer Avocado, which indexes text data for "subsequent search" [8]. In STRIDE all clinical documents and reports are full-text indexed and searchable using Oracle Text [14]. OpenMRS uses Apache Lucene with Hibernate Search to perform a full text indexing on all registered entities.[2] In tranSMART all files and table records are indexed and can be queried via Apache Solr [13].

The wide-spread open source CDW i2b2 [11] just offers the SQL *like*-operator[3], that can be used to perform wildcard queries.

There are also document oriented CDWs like Roogle, which stores clinical texts and their metadata [1]. The system extracts clinical concepts from these texts, but they are predefined, like e.g. the MeSH thesaurus. IE in clinical documents is complicated by the fact, that many phrases are negated [18]. Therefore it is important to identify these negations and exclude them from the queried results. The full-text search-engine based CDW Dr. Warehouse improves its functionality by identifying negations and family history context. Texts are divided into subtexts, which are indexed, so it is possible to query the text parts, which contain affirmed findings of a patient. They used ConText (see below) to find these subtexts. That system is built for French texts [2].

However, ad hoc information extraction with the option to count medical concepts and their numeric values has not yet been described. Ad-hoc IE means the technical concept of extracting the existence of any concept (e.g. chronic kidney disease) or any number (e.g. the LVEF value) from a source in real-time thus allowing the application of the usual query operations (e.g. counting the number of patient cases with LVEF < 45) on the extracted concepts.

1.2 Negation Detection

A prerequisite for useful counts of search results in clinical texts is the reliable detection of negations. Chapman et al. al introduced in 2001 the NegEx algorithm for identifying negated findings in discharge summaries [19]. Chapman et al. extended it to ConText, which analyses whether the clinical condition is negated, hypothetical, historical or concerns another person than the patient himself. Moreover the scope determination was changed from a fixed size of six tokens to the next trigger token in the sentence [20]. The trigger sets have been translated in multiple languages: Swedish [21], French [22], Spanish [23, 24], Dutch [25], Swedish, French and German [26]. The German triggers have been extended and the algorithm has been adapted. It showed good results in a small evaluation with eight discharge letters (F1-score: 0.91) and 175 clinical notes (F1-score 0.96) [27]. One evaluation on negation detection in German clinical text was made by Gros and Stede with their Netopus system [28]. It achieved good results on finding the negation triggers, but could only determine the exact scope in 54% in German medical texts.

Other approaches as the popular token-based algorithm NegEx exist in particular for English texts. Rule-based systems use ontologies [29] or syntactic parsing [30]. Dependency parsing was used as well to enrich the negation detection and scope determination [31]. Even some machine learning approaches were made e.g. by trying to classify a negation with a support vector machine [32]. A good overview is given by Mehrabi [33]. Although many papers show good results, Wue et al. show that the negation detection problem is not solved yet. If no in-domain development or training-data is available the algorithms perform poor [32].

2. Objectives

The goal is to develop a pipeline for ad hoc IE being able to reliably count Boolean and constrained numerical values in clinical textual documents. Because many findings are negated in clinical texts, this includes three subtasks: (a) recognizing and excluding negations and their scopes in text documents, (b) extracting Boolean and (c) numeric values fulfilling constraints (e.g. LVEF < 45) with context sensitive search queries.

This ad hoc IE shall not be considered as a replacement for conventional IE, but rather a supplement allowing quick shallow data aggregation to potentially answer any question in the first approximation without the complex pre-defined specifications required for standard IE.

3. Methods

3.1 PaDaWaN

In the document oriented CDW system PaDaWaN [34] every text, like e.g. a discharge letter, is analyzed within an analysis pipeline and stored in the index of Apache Solr server, a popular full-text search engine built on top of the index library Apache Lucene. Afterwards the text can be queried from physicians in the PaDaWaN-Web GUI [34].

PaDaWaN is implemented at the University Hospital of Würzburg, including various data types integrated from various medical domains and a privacy protection concept approved by the institution's data protection officer. It uses state of the art techniques such as de-identification of text contents and pseudonymization.

3.2 System Design of Text Search Extension

During data integration, the texts are preprocessed in an analysis pipeline. In addi-

2 https://wiki.openmrs.org/pages/viewpage.action?pageId=15139564

3 http://community.i2b2.org/wiki/display/DevForum/Text+search+in+i2b2

tion to standard NLP tasks, such as tokenizing, stemming and stop-word removal, this pipeline also includes the special function negation detection. All negated parts with their scopes are identified and removed. The remaining text with only affirmed and no negated findings and the original text are both passed in the pipeline. At the end, like all other information, they are stored in the index so that they can be queried separately afterwards.

During run time, the index can be requested through the interface. That's where the ad hoc IE takes place. To make that possible we developed query features and output features to extract information to make them available for further processing [35].

3.3 Extended NegEx-Algorithm

We used an extended version of the NegEx-algorithm to identify the negations. Some adaptions were made, because the input of the root algorithm has two arguments: a sentence and a concept token. The output is whether the concept is negated or not. In our system, negation detection is part of the preprocessing, but the concepts to be classified are only queried later at runtime. Therefore, we determined the negations and their scope within a sentence. All concepts, that are located in a negated scope, are considered as negated. Like ConText, we extended the NegEx algorithm by changing the negation scope from fixed length of six tokens to a variable length to the next trigger token. Moreover we extended the trigger token set.

3.3.1 Trigger Set

We took the trigger list from [27], which is based on the translation to German by [26]. We edited and extended this list to a size of 548 triggers.[4] While Cotik et al. [27] label the trigger tokens in a sentence according to a fixed precedence list, starting with pre-negating trigger tokens (PREN), follow by post-negated trigger tokens (POST), prepositions (PREP) and pseudo-negating trigger tokens (PSEU), we always choose the label with the longest match sequence. E.g. the tokens *"keine Anhaltspunkte für"* (no evidence for) are assigned to the pre-negating label (PREN) while the tokens of the subsequence "*keine Anhaltspunkte*" (no evidence) are assigned to the post-negating label (POST). See ▶Table 1.

Table 1 Example for pre and post negating triggers.

PREN	**Keine Anhaltspunkte für** pulmonale Metastasierung.
POST	Für eine pulmonale Metastasierung ergaben sich **keine Anhaltspunkte**.

3.3.2 Algorithm Description

Input: sentence, trigger list.
Output: negated scopes in the sentence.

1. All trigger tokens in a given sentence are annotated with their corresponding label.
2. The algorithm iterates over the trigger tokens of the sentence. At every post-negating trigger token a negation scope is added from the last trigger token (or begin of the sentence) to the current one. At every pre-negating trigger token a negation scope is added from the current trigger token to the next trigger token (or the end of the sentence).

3.3.3 Sentence Splitting

As mentioned above, the input of NegEx is one sentence. Because we had to process an entire text, sentence spitting had to be applied. The text is split up at the usual punctuations excluding punctuations in abbreviations, dates and blocks in parentheses. The comma sign was added to the split-token-list, since in many clinical abbreviated texts it serves as a regular period, e.g. *"Zungenmotilität normal, keine Zungenfibrillationen, Zungenkraft normal" (Tongue motility normal, no tongue fibrillation, tongue force normal).* Some exceptions were made to capture Hearst-patterns, i.e. enumerations like *"Keine Stauungszeichen, Infiltrate oder Ergüsse."(No cramps, infiltrates or effusions.).* That step is performed in the Apache UIMA[5] pipeline.

Example 1: Echocardiogram reports
Kein *Pleuraerguss*, ***kein*** *konfluierendes Infiltrat*, ***keine*** *Stauungszeichen*. ***Keine*** *malignitätssuspekten Rundherde. Herz links betont vergrößert. Aorta elongiert und sklerosiert.*

Example 2: Urethrocystoskopie
Harnröhre zeigt ***keinen Hinweis*** *für eine Striktur, Prostata ist* ***nicht*** *obstruktiv, nebenbefundlich enger Blasenhals, Blasenschleimhaut trabekuliert, jedoch* ***kein Hinweis*** *für einen exophytischen Blasentumor. Ostien bds. orthotop, schlitzförmig mit klarem Urinjet.*

The negation triggers are bold and the negation scope is underlined. First, the text is split up and then the algorithm is applied. For further processing in our analysis pipeline, the negated parts are removed from the text. As a result, only affirmed findings are included in the text.

3.4 CDW Integration

The negation detection identifies all negated parts in a text and removes them from the text. The remaining text with no negated findings and the original text are both added to the PaDaWaN, an index based CDW. The texts and all other information in the DW, like core data, ICD10 diagnosis, laboratory findings, procedures and other report findings, can be queried through a web-GUI by physicians.

3.5 Query Features

Because PaDaWaN is a search-engine-based CDW, it offers many query features for texts [35]. We developed and extended these features so that users can create queries at runtime that recognize any concepts in texts and extract their values on the fly. There are well known functions like Boolean retrieval, wildcards and phrase queries, and more advanced features like a context specific query, a regular expression query with filter options and output definition.

4 The list is available at: go.uniwue.de/padawan

5 https://uima.apache.org/

Table 2 Example for a context-sensitive query.

Query	Matching Text	
[dilatiert Vorhof]	*Der linke Vorhof ist deutlich dilatiert*	(4)
[3+ Suffiziente Mitralklappe]	*Suffiziente Aorten- und Mitralklappe*	(5)

Table 3 Example of the regular expression feature for querying (6), constraining (7) and extracting (8) a numeric concept (Puls = pulse, ZAHL = NUMBER). "$1" is a reference to the extracted concept (the first expression in round parentheses or its equivalent predefined class, i.e. "ZAHL").

Syntax	Alternative Syntax	
/Puls ZAHL/	*/Puls [0–9]+/*	(6)
/Puls ZAHL/[ZAHL > 150]	*/Puls ([0–9]+)/[$1 > 150]*	(7)
/Puls ZAHL/[ZAHL > 150] ZAHL	*/Puls ([0–9]+)/[$1 > 150] $1*	(8)

Boolean retrieval filters texts, that contain the given query tokens. They can be combined via logical operators *'and', 'or', 'not'.* A query for heart failure could look like:

(cardiac decompensation) OR (heart failure) (1)

A wildcard character is used to substitute any characters in a word, e.g. in the German compound words like (2). (mitral insufficiency)

*Mitral*insuffizienz* (2)

A phrase query matches texts containing a particular sequence of words. The entire sequence must match like:

"diabetes mellitus" (3)

In contrast to a Boolean query, where the terms can be anywhere in the text, in a context-sensitive query the user has control over the proximity and order of the terms. Both are adjustable. The given terms must occur in the same sentence (see ▶Table 2). The query (4) would match any text that contains these two terms in one sentence with not more than seven words (default value) between them. The order of words does not matter. In contrast to the query (5), here the order of words matters and the gap between the query tokens must not be more than three words.

The regular expression (regex) query feature is a further function to filter texts. Experienced user can write a regular expression using the standard regular expression syntax with predefined character classes, quantifiers, alternatives and grouping. The regular expression is defined between slashes (see ▶Table 3). For users with no computer science background we added predefined classes for convenience, like ZAHL (number), which are compiled to a regular expression automatically.

▶Table 3 shows an example of the regex query feature for a numeric concept. Line (6) queries the existence of the concept in the text. Line (7) adds a numeric condition, which is defined in brackets. That query would match all texts with the token *Puls* followed by a number, that is bigger than 150. This number would be returned within the result. Line (8) extracts the numeric value of the concept for further computation. That is defined in the query syntax by writing the desired group (ZAHL or "$1") at the end of the query. If a query is run with that extraction mode, the engine returns a list with all type safe extracted numbers for the queried concept.

Further features of the query syntax are given in the example 9–10.

/Blutdruck ZAHL\/ZAHL/[$2 >150] (9)

/([0–9]+)\.([0–9]+)\.([0–9]+)/$3-$2-$1 (10)

If the query contains more than one number like *'Blutdruck 150/90'* (blood pressure), the numbers can be referenced using the $-notation (see examples 8 and 9). The escape character is the backslash. Not only can the predefined class NUMBER be referenced and extracted, but also self-created regex groups can be used. The groups are defined in parentheses in accordance with the regex syntax (used in lines 6, 7, 8, 9 and 10).

As an additional use case example, the runtime IE mechanism can be used to transform notations like a German date (i.e. "dd.MM.yyyy") into the English equivalent ("yyyy-MM-dd") for further computation (see 10).

The regex query, containing a numeric condition, is compiled into the regular expression. (11) is the compiled result of query (7). This regular expression is passed to the index server as a normal constraint query and can be evaluated efficiently. So no post-processing of the results is necessary.

Puls (15[1–9])|(1[6–9][0–9]{1,})|([2–9][0–9]{2,})|([1–9][0–9]{3,}) (11)

This paper evaluated the functionality of these query features that we developed and extended. The usability in a user interface of a CDW by e.g. by physicians was not part of this work (see conclusion and further work).

3.6 Evaluation

All evaluations were made by randomly selecting texts out of the PaDaWaN CDW. To protect privacy, these texts are de-identified and in addition must not leave the clinical network.

3.6.1 Negation Detection

For the evaluation of the negation detection experiments we took two different domains. The first domain was chest X-ray reports. Their text structure is a telegraphic style with short sentences, mostly containing noun phrases.

We created a manually annotated gold standard of 100 reports. First, the texts were automatically pre-annotated to save time, using an information-extraction terminology created by physicians [36]. Afterwards these texts were manually corrected in the ATHEN environment[6] to achieve the gold standard.

6 http://www.is.informatik.uni-wuerzburg.de/research_tools_download/athen/

In contrast to chest X-ray reports, we used a second domain with a more complex sentence structure and a larger vocabulary, i.e. discharge letters. We created a gold standard as well for 50 letters. That procedure was similar to the chest X-ray gold standard, but because we had no terminology for an entire discharge letter, we tried to identify findings and medical concepts in the texts. Therefore we took the German list of Alpha-ID[7], a list with more than 80.000 diagnoses, and the German version of MeSH (Medical Subject Headings)[8], a list with more than 60.000 medical concepts. Additionally we used a part-of-speech tagger [37] to label all named entities and nouns with no lemma. That are nouns, which are unknown to the tagger; these are technical terms of a specific domain, in this case: mostly medical concepts. Next, we used our extended NegEx-algorithm to label negation trigger and their span. This pre-annotated data was again corrected manually to create a gold standard.

3.6.2 Ad Hoc Information Extraction

The ad hoc information extraction was evaluated in two domains as well: echocardiogram reports and discharge letters. We randomly picked 1000 texts from each domain. The ad hoc IE queries were run in the PaDaWaN-system and the results were manually evaluated.

The medical concepts and their synonyms in query syntax like (10) are the input for the Boolean ad hoc information extraction (Engl. mild mitral insufficiency):

leicht Mitral*insuffizienz* (10)

Similar, regular expressions describing the medical concept and the value to be extracted are the input for the numeric ad hoc information extraction:

/(Cholesterin|Chol)(\.)?(:)? ([0–9]+) mg/ $4 (11)

The PaDaWaN-system used an Apache Solr 7.0 server out-of-the-box and it was run with one instance, two nodes and two shards. Caching was disabled during the tests.

7 https://www.dimdi.de/static/de/klassi/alpha-id/

8 https://www.dimdi.de/static/de/klassi/mesh_umls/mesh/

Table 4 Performance of the negation detection of medical concepts in the two domains.

	Chest X-ray	Discharge letter
Documents	100	50
Negations	619	397
TP	608	366
FP	1	1
FN	11	31
Precision	0.998	0.997
Recall	0.982	0.922
F1	0.990	0.958

Table 5 Error analysis of wrong classified concepts in the negation detection.

	Chest X-ray	Discharge letter	Combined
Sentence splitting	3 (0.25)	2 (0.06)	5 (0.12)
Wrong documentation	1 (0.08)	0 (0.00)	1 (0.02)
Complex sentence structure	3 (0.25)	5 (0.16)	8 (0.19)
Missing negation triggers	5 (0.42)	24 (0.77)	29 (0.67)

Table 6 Performance of the retrieval of the negated scopes and their length in discharge letters.

Scope retrieval						Scope length		
TP	FP	FN	Precision	Recall	F1	Exact	Too Narrow	Too Wide
348	4	6	0.989	0.983	0.986	318 (0.91)	2 (0.01)	28 (0.08)

4. Results

4.1 Negation Detection

The F1-score for the negation detection was 0.99 in the telegraphic-style chest X-ray reports and 0.96 in the more complex discharge letters. ▶Table 4 shows the detailed results of the evaluation of the negation detection of medical concepts. (TP = true positives, FP = false positives, FN = false negatives).

While the precision with just one false positive in each domain is high, the recall contained some false negatives, which refer to different error sources. In 67% of all errors the negation triggers were not contained in the trigger set. This was especially the main problem for discharge letters (77% of its errors). Due to the natural flow of speech the variety of the negation trigger was much greater than in the chest X-ray reports. This explains the difference in the overall performance between the two domains: F1-scores: chest X-ray 0.99, discharge letter 0.96. ▶Table 5 summarizes a categorization of the error sources.

The ten most common missing negation triggers are: kein Anhalt für (no clue for), nicht erforderlich (not mandatory), nicht angegeben (not specified), keine Notwendigkeit (no need), nicht anzuraten (not recommended), nicht bekannt (not known), nicht mehr nachweisbar (no longer detectable), nicht tastbar (not palpable), traten nicht mehr auf (did not occur anymore), keine Indikation (no indication).

The negated scopes were detected with a F1-score of 0.97. The exact length was determined in 91%. ▶Table 6 shows the detailed results in the retrieval of the negated scopes and the determination of their length in the discharge letter domain.

	Discharge letter
Sentence splitting	8 (0.28)
Documentation fault	3 (0.10)
Complex sentence structure	5 (0.17)
Wrong labeling of filler words	6 (0.21)
Other errors	7 (0.24)

Table 7 Error analysis of wrongly determined negation scope in discharge letters.

Table 8 Performance of Boolean ad hoc information extraction using the context sensitive query feature for medical concepts: (1) mild mitral insufficiency, (2) high mitral insufficiency and (3) mild aortic stenosis.

		Dataset	FP	FN	TP	Recall	Precision	F1
(1)	Leichtgradige Mitralinsuffizienz	echocardiography	0	7	304	0.977	1	0.987
(2)	Hochgradige Mitralinsuffizienz	echocardiography	0	0	14	1	1	1
(3)	Leichtgradige Aortenstenose	echocardiography	0	3	160	0.982	1	0.991

Table 9 Performance of numeric ad hoc information extraction using the regex query feature for the medical concepts: (1) Cholesterol, (2) Glucose, (3) BMI, (4) LVEF, and (5) age.

		Dataset	FP	FN	TP	Recall	Precision	F1
(1)	Cholesterin	discharge letter	0	2	158	0.988	1	0.994
(2)	Glucose	discharge letter	0	6	336	0.982	1	0.991
(3)	BMI	discharge letter	0	0	44	1	1	1
(4)	LVEF	echocardiography report	6	0	452	1	0.987	0.993
(5)	age	discharge letter	136	4	49	0,93	0,27	0,41

Many errors in the chest X-ray report were made by splitting the sentence at wrong positions. The reasons are: Some abbreviations were unknown, the corresponding period was misinterpreted. Furthermore enumerations were not recognized and also some filler words like *and, too* or commas were mistakenly included at the end of negations scope. ▶Table 7 summarizes a categorization of the error sources.

4.2 Boolean Ad Hoc Information Extraction

The evaluation of the Boolean ad hoc information extraction in the heart echo documents showed good results with a F1-score between 0.98 and 1 (see ▶Table 8). Three concepts with modifiers were queried in 1000 chest echocardiography reports. Every query contained synonyms and wildcards to match the concepts in the texts. The context sensitive query feature was used to ensure that the queried tokens relate to each other.

All errors refer to an incorrect sentence splitting in the preprocessing.

The average processing time was 72 ms to query the hit count and 2.8 s to export all extracted information.

4.3 Numeric Ad Hoc Information Extraction

▶Table 9 shows the result of numeric ad hoc information using the regex query feature with examples from two datasets. Some regex-queries (cholesterol, glucose, age) contained synonyms of the concepts and all queries accepted multiple notations. All F1-scores except "age" are above 0.99.

The last concept "age" is a difficult task, because it not only refers to the current age of the patient but also to his or her history or to other persons than the patient. We achieved a high recall, but a low precision (see ▶Table 9, error analysis see ▶Table 10). However, extracting the age of a person is not necessary in practice, because it is accessible as structured data.

The error analysis revealed that the eight false negatives for the first four concepts in ▶Table 9 are caused by an incorrect sentence splitting in the preprocessing, while the six false positives result from incorrect recognition of intervals instead of single numbers.

For extracting the concept *age* in the discharge letter, 97% of the errors refer to the wrong context, as it can be seen in ▶Table 10. The errors are subdivided in four parts: age in the patient history ("First occurrence at the age of 30 years."), age in family history ("The grandmother died with 87 years."), relative years in the patient history ("5 years ago"), relative years to future events ("next examination in 2 years"). They can be grouping errors in the patient history (73%), in the family history (11%) and future events (13%).

The average time was 1075 ms to query the number hits and 1536 ms to export all extracted information.

5. Discussion

5.1 Negation Detection

The negation detection performed very well in both domains and slightly better as Cotik et al. They achieved a F1-score of 0.96 score on clinical notes and 0.91 on discharge letters (see ▶Table 11) [27].

In 91% of all detected negations scopes, the length of the scope was determined correctly. That score is lower, but keep in mind, that the F1-score of negated concepts was quite good. So some of these miscalculated scopes did not contain relevant information or clinical concepts. In fact, the determination is a difficult task, Gros and Stede could only compute in 54% the

exact scope in medical texts (see ▶Table 12) [28].

5.2 Boolean Ad Hoc Information Extraction

The extraction for Boolean values using the context sensitive query worked well. The text is split up into sentences and logical parts and negations are removed at preprocessing time. Afterwards queries can be run against the index. The tokens of a query must match the tokens in one sentence. Wildcards in the query tokens match many variants of word spellings. That simple mechanism is a very powerful tool. Even the span-limitation-feature between the words is often not necessary. The F1-scores between 0.99 and 1.0 confirm that approach.

5.3 Numeric Ad Hoc Information Extraction

The regular expression queries for the numeric ad hoc information extraction provided very good results as well. But they showed the limitation of that approach, too. The extraction of the desired values works fine, but the context must be clear. If the concept always refers to the patient, the regex query is a powerful feature as well, which extracted values with a F1-score bigger than 0.99. Currently we work towards recognizing different contexts of concepts (e.g. age of relatives).

Table 10 Error analysis of ad hoc information extraction of the concept age in the entire discharge letter.

	Number of Errors	Percentage
Age in patient history	18	0.13
Age in family history	15	0.11
Relative years in patient history	81	0.60
Relative years to future events	18	0.13
Unexpected syntax	4	0.03

Table 11 Comparison of F1-scores to other negation detection approaches for German clinical texts.

Data set	Cotik et al.	Data set	Our approach
Discharge letter	0.91	Discharge letter	0.96
Clinical notes	0.96	Chest X-ray	0.99

Table 12 Comparison of F1-scores to other negation scope length determination approaches for German clinical texts.

	Gros and Stede	Our approach
Data set	cardiology report	discharge letters
Exact	0.54	0.91
Too narrow	0.34	0.01
Too wide	0.12	0.08

Table 13 Comparison between ad hoc information extraction and standard IE.

	Ad hoc IE	Standard IE
Scope	specific concept	entire domain
Development effort	low	high
Promptness	fast	slow
Adaptability by user	yes	no
Accuracy	lower	higher
Confidence	low	high

5.4 Ad Hoc IE Versus Standard IE

Ad hoc IE has several advantages in comparison to standard IE, but also some shortcomings. Its advantages are the low development effort, the promptness of the results and the adaptability by the user to his or her particular question. The main disadvantage is that the accuracy of the results is usually lower and there are no evaluation results available resulting in a lower confidence. The biggest difference however is the development effort, which is very low for Ad hoc IE (just entering the concept with its variants) and high for standard IE requiring the definition of a terminology and learning or engineering the extraction patterns. ▶Table 13 summarizes the comparison. It would be attractive to integrate concepts from the ad hoc IE into the permanent part of the CDW by enriching its catalog of concepts.

5.5 PaDaWaN Versus Other CDWs

This approach is novel and exceeds existing systems: Roogle [1] extracts predefined concepts from texts at preprocessing and makes them retrievable at runtime.

Dr. Warehouse [2] applies negation detection and indexes the produced subtexts which include affirmed findings.

The introduced approach combines negation detection and the extraction of concepts, but not in an unmodifiable way that has only a fixed set of concepts and that takes place at the time of preprocessing. We provide ad hoc, dynamic, interactive and adjustable information extraction of random concepts and even their values on the fly at runtime.

6. Conclusion and Further Work

A pipeline for a negation sensitive ad hoc information extraction of Boolean and numeric concepts was developed and evaluated allowing context sensitive and regular expression queries for texts within a CDW.

We have shown that an ad hoc IE can deliver good results. Since it can be used interactively and customized by users at

runtime, it is a good complement to existing IE approaches.

It is intended, that physicians can make direct use of the ad hoc information extraction in the user interface of a CDW. So the next step is the development and evaluation of a smart user interface including the above mentioned extension of the catalog of concepts.

Acknowledgment

The authors thank Vincent Truchseß for his annotation help.

References

1. Cuggia M, Garcelon N, Campillo-Gimenez B, Bernicot T, Laurent J-F, Garin E, et al. Roogle: an information retrieval engine for clinical data warehouse. Stud Health Technol Inform 2011; 169: 584–588.
2. Garcelon N, Neuraz A, Benoit V, Salomon R, Burgun A. Improving a full-text search engine: the importance of negation detection and family history context to identify cases in a biomedical data warehouse. J Am Med Inform Assoc 2017; 24(3): 607–613.
3. Starlinger J, Kittner M, Blankenstein O, Leser U. How to improve information extraction from German medical records. it-Information Technology 2016; 58(10).
4. Krieger H-U, Spurk C, Uszkoreit H, Xu F, Zhang Y, Müller F, et al. Information Extraction from German Patient Records via Hybrid Parsing and Relation Extraction Strategies. In: Chair; NCCChoukri K, Declerck T, Loftsson H, Maegaard B, Mariani J, Moreno A, Odijk J, Piperidis S, et al., editors. Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14) Reykjavik, Iceland: European Language Resources Association (ELRA); 2014. p. 2043–2048.
5. Toepfer M, Corovic H, Fette G, Klügl P, Störk S, Puppe F. Fine-grained information extraction from German transthoracic echocardiography reports. BMC Med Inform Decis Mak 2015; 15: 91.
6. Dorda W, Wrba T, Duftschmid G, Sachs P, Gall W, Rehnelt C, et al. ArchiMed: a medical information and retrieval system. Methods Inf Med 1999; 38(1): 16–24.
7. Gabetta M, Limongelli I, Rizzo E, Riva A, Segagni D, Bellazzi R. BigQ: a NoSQL based framework to handle genomic variants in i2b2. BMC Bioinformatics 2015; 16(1): 415.
8. Hu H, Correll M, Kvecher L, Osmond M, Clark J, Bekhash A, et al. DW4TR: a data warehouse for translational research. J Biomed Inform 2011; 44(6): 1004–1019.
9. Hussain S, Ouagne D, Sadou E, Dart T, Jaulent M-C, De Vloed B, et al. EHR4CR: A Semantic Web Based Interoperability Approach for Reusing Electronic Healthcare Records in Protocol Feasibility Studies. In: SWAT4LS. 2012.
10. Pennington JW, Ruth B, Italia MJ, Miller J, Wrazien S, Loutrel JG, et al. Harvest: an open platform for developing web-based biomedical data discovery and reporting applications. Journal of the American Medical Informatics Association 2013; 21(2): 379–383.
11. Murphy SN, Weber G, Mendis M, Gainer V, Chueh HC, Churchill S, et al. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). Journal of the American Medical Informatics Association 2010; 17(2): 124–130.
12. Wolfe BA, Mamlin BW, Biondich PG, Fraser HS, Jazayeri D, Allen C, et al. The OpenMRS system: collaborating toward an open source EMR for developing countries. In: AMIA Annual Symposium Proceedings. American Medical Informatics Association; 2006. p. 1146.
13. Harris PA, Taylor R, Thielke R, Payne J, Gonzalez N, Conde JG. Research electronic data capture (REDCap)—a metadata-driven methodology and workflow process for providing translational research informatics support. Journal of Biomedical Informatics 2009; 42(2): 377–381.
14. Lowe HJ, Ferris TA, Hernandez PM, Weber SC. STRIDE–An integrated standards-based translational research informatics platform. In: AMIA Annual Symposium Proceedings. American Medical Informatics Association; 2009. p. 391.
15. Canuel V, Rance B, Avillach P, Degoulet P, Burgun A. Translational research platforms integrating clinical and omics data: a review of publicly available solutions. Briefings in Bioinformatics 2014; 16(2): 280–290.
16. Danciu I, Cowan JD, Basford M, Wang X, Saip A, Osgood S, et al. Secondary use of clinical data: the Vanderbilt approach. Journal of Biomedical Informatics 2014; 52: 28–35.
17. Dziuballe P, Forster C, Breil B, Thiemann V, Fritz F, Lechtenbörger J, et al. The single source architecture x4T to connect medical documentation and clinical research. Stud Health Technol Inform 2011; 169: 902–906.
18. Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. Evaluation of negation phrases in narrative clinical reports. In: Proceedings of the AMIA Symposium. American Medical Informatics Association; 2001. p. 105.
19. Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. A simple algorithm for identifying negated findings and diseases in discharge summaries. Journal of Biomedical Informatics 2001; 34(5): 301–310.
20. Harkema H, Dowling JN, Thornblade T, Chapman WW. ConText: an algorithm for determining negation, experiencer, and temporal status from clinical reports. Journal of Biomedical Informatics 2009; 42(5): 839–851.
21. Skeppstedt M. Negation detection in Swedish clinical text: An adaption of NegEx to Swedish. Journal of Biomedical Semantics. 2011; 2(3): S3.
22. Deléger L, Grouin C. Detecting negation of medical problems in French clinical notes. In: Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium. ACM; 2012. p. 697–702.
23. Cotik V, Stricker V, Vivaldi J, Rodriguez H. Syntactic methods for negation detection in radiology reports in Spanish. ACL 2016. 2016. p. 156.
24. Costumero R, López F, Gonzalo-Mart'ın C, Millan M, Menasalvas E. An approach to detect negation on medical documents in Spanish. In: International Conference on Brain Informatics and Health. Springer; 2014. p. 366–375.
25. Afzal Z, Pons E, Kang N, Sturkenboom MC, Schuemie MJ, Kors JA. ContextD: an algorithm to identify contextual properties of medical terms in a Dutch clinical corpus. BMC Bioinformatics 2014; 15(1): 373.
26. Chapman WW, Hilert D, Velupillai S, Kvist M, Skeppstedt M, Chapman BE, et al. Extending the NegEx lexicon for multiple languages. Stud Health Technol Inf 2013; 192: 677.
27. Cotik V, Roller R, Xu F, Uszkoreit H, BuddeO K, SchmidtO D. Negation Detection in Clinical Reports Written in German. BioTxtM 2016. 2016. p. 115.
28. Gros O, Stede M. Determining Negation Scope in German and English Medical Diagnoses. In: Taboada M, Trnavac R, editors. Nonveridicality and Evaluation: Theoretical, Computational and Corpus Approaches. (Studies in Pragmatics 11). Leiden/Boston: Brill, 2013. p. 113–126.
29. Elkin PL, Brown SH, Bauer BA, Husser CS, Carruth W, Bergstrom LR, et al. A controlled trial of automated classification of negation from clinical notes. BMC Medical Informatics and Decision Making 2005; 5(1): 13.
30. Huang Y, Lowe HJ. A novel hybrid approach to automated negation detection in clinical radiology reports. Journal of the American Medical Informatics Association 2007; 14(3): 304–311.
31. Sohn S, Wu S, Chute CG. Dependency parser-based negation detection in clinical narratives. AMIA Jt Summits Transl Sci Proc 2012; 2012: 1–8.
32. Wu S, Miller T, Masanz J, Coarr M, Halgrim S, Carrell D, et al. Negation's not solved: generalizability versus optimizability in clinical natural language processing. PloS One 2014; 9(11): e112774.
33. Mehrabi S, Krishnan A, Sohn S, Roch AM, Schmidt H, Kesterson J, et al. DEEPEN: A negation detection system for clinical text incorporating dependency relation into NegEx. Journal of Biomedical Informatics 2015; 54: 213–219.
34. Dietrich G, Fell F, Fette G, Krebs J, Ertl M, Kaspar M, et al. Web-PaDaWaN: Eine Web-basierte Benutzeroberfläche für ein klinisches Data Warehouse. In: HEC 2016, Joint Conference of GMDS, DGEpi, IEA-EEF, EFMI, DocAbstr 421, 2016. 2016.
35. Dietrich G, Ertl M, Fette G, Kaspar M, Krebs J, Mackenrodt D, et al. Extending the Query Language of a Data Warehouse for Patient Recruitment. Stud Health Technol Inform 2017; 243: 152.
36. Krebs J, Corovic H, Dietrich G, Ertl M, Fette G, Kaspar M, et al. Semi-automatic Terminology Generation for Information Extraction from German Chest X-ray Reports. Stud Health Technol Inform 2017; 243: 80.
37. Schmid H. Probabilistic part-of-speech tagging using decision trees. In: New methods in language processing. 2013. p. 154.