# CodeMapper: Semi-automatic coding of case definitions

Benedikt Becker\*1, Erik M. van Mulligen1, Daniel Weibel1, Paul Avillach1,2, Miriam Sturkenboom1,3, Jan A. Kors1

1 Department of Medical Informatics, Erasmus University Medical Center, Rotterdam, The Netherlands  
2 Department of Biomedical Informatics, Harvard Medical School, Boston, United States  
3 Department of Epidemiology, Erasmus University Medical Center, Rotterdam, The Netherlands  
\* Corresponding author. Address: Department of Medical Informatics, Erasmus University Medical Center, P.O. 21455, 3001 AL Rotterdam, The Netherlands. Email: b.becker@erasmusmc.nl

##### Document info

Target journal: *Pharmacoepidemiology and drug safety*  
Word cound: 3,910

##### History

|  |  |  |
| --- | --- | --- |
| Benedikt Becker | January 5, 2016 | Draft v1 |
| Miriam Sturkenboom | February 7, 2016 | Comments & suggestions |
| Benedikt Becker | February 17, 2016 | Draft v2 |
| Paul Avillach | February 29, 2016 | Comments & suggestions |
| Jan Kors | March 9, 2016 | Commets & suggestions |
| Benedikt Becker | April 3, 2016 | Draft v3 |

##### TODO

* format references
* slant *CodeMapper*?
* s/coding system/(biomedical) vocabulary/ ?

# Introduction

##### Background

Information from multiple electronic health record (EHR) databases is often combined in collaborative epidemiological studies to improve sample size and heterogeneity and hence the predictive power of the study [Trifiro2014]. However, EHR databases in Europe use different coding systems to describe medical information, such as the International Classification of Diseases (ICD), the International Classification of Primary Care (ICPC), or READ codes. The extraction of an event then requires usually several steps to achieve consistency between different databases. First, the event is described textually by a case definition in the study protocol. The textual description is then mapped for each coding system into a set of codes that represents the case definition. Subsequently, the code sets are converted into queries to identify cases or phenotype the event in EHR databases [Overby2013b]. The code sets and queries are then iteratively harmonized between databases by comparison with benchmarks from the literature and by feedback from the database custodians. This harmonizing between databases was first described for the EU-ADR project [Avillach2009,Coloma2011,Avillach2013], and later on extended by a common data model (OMOP [Overhage2012,Reich2012,Gini2016]) and component algorithms (EMIF [EMIF2015,Gini2016]).

Currently, the mapping of the textual case definitions into code sets for each coding system is largely a manual process. Given the number and complexity of the involved coding systems, the mapping process can pose an important bottleneck to the rapid implementation of collaborative epidemiological studies. Furthermore, the rational for including or excluding individual codes in the mapping is often not documented in the process, which can hamper the adequate re-use of code sets and queries in subsequent studies.

##### Objective

We present a web application called CodeMapper, which assists in mapping cases definitions in code sets from different coding systems while keeping a transparent record of the complete mapping process. We evaluate the application by comparing code sets that were automatically generated by CodeMapper with reference code sets that were manually created.

# Methods

In this section, we will first give a short description of the user interface of CodeMapper, provide then details about its implementation, and finally present how the approach was evaluated.

##### The user interface

The CodeMapper application consists of three screens. On the first screen, the user can enter a clinical case definition of an event as free-text. Medical concepts are automatically identified in the text and highlighted. Concepts representing medical disorders are preselected for further processing in the application, but the user can select and deselect any identified concepts depending on their relevance for the described event. The set of concepts defines a mapping in CodeMapper by the codes that are associated with the concepts. It can be further revised on the second screen.

On the second screen, the mapping is displayed as a table with one row for each medical concept, and one column for each coding system (a screenshot of the application is shown in figure 1). Each cell contains the codes that are used in the coding system in the column to describe the concept in the row. Several operations are available for revising the mapping. or expand concepts to broader and narrower concepts and select them for inclusion in the mapping. For example, expanding the concept *Hepatitis* to narrower concepts provides concepts for the subtypes of the disease, such as *Hepatitis A*, *Hepatitis B,* or *Hepatitis C*. Expanding it to broader concepts provides categories such as *Liver disease* or *Infection*. The expansion of concepts can be used in this way to improve the sensitivity of the mapping. Talso add or remove that should be targeted by the mappingcodes updated and displayed in the table Every operation is recorded in a history for later traceability of the mapping process. Comments can be attached to the concepts of a mapping to collect feedback about the mapping. The user can store the mapping at any point online for archiving, which makes the mapping also available to other users of the application. In order to save the mapping, the user is required to provide a short summary of her or his modifications, which is incorporated into the history of the mapping to provide additional context to the changes. Finally, the user can also download the mapping as a spreadsheet document, for example incorporating the codes into extraction queries. The downloaded document comprises the original free-text case definition, the concepts of the mapping, the code sets for each coding system, and the full history of the mapping process. The third screen of CodeMapper shows the history of user interaction that resulted in the current mapping.

CodeMapper is implemented as a web application and freely available for non-commercial use at <https://euadr.erasmusmc.nl/CoMap>.

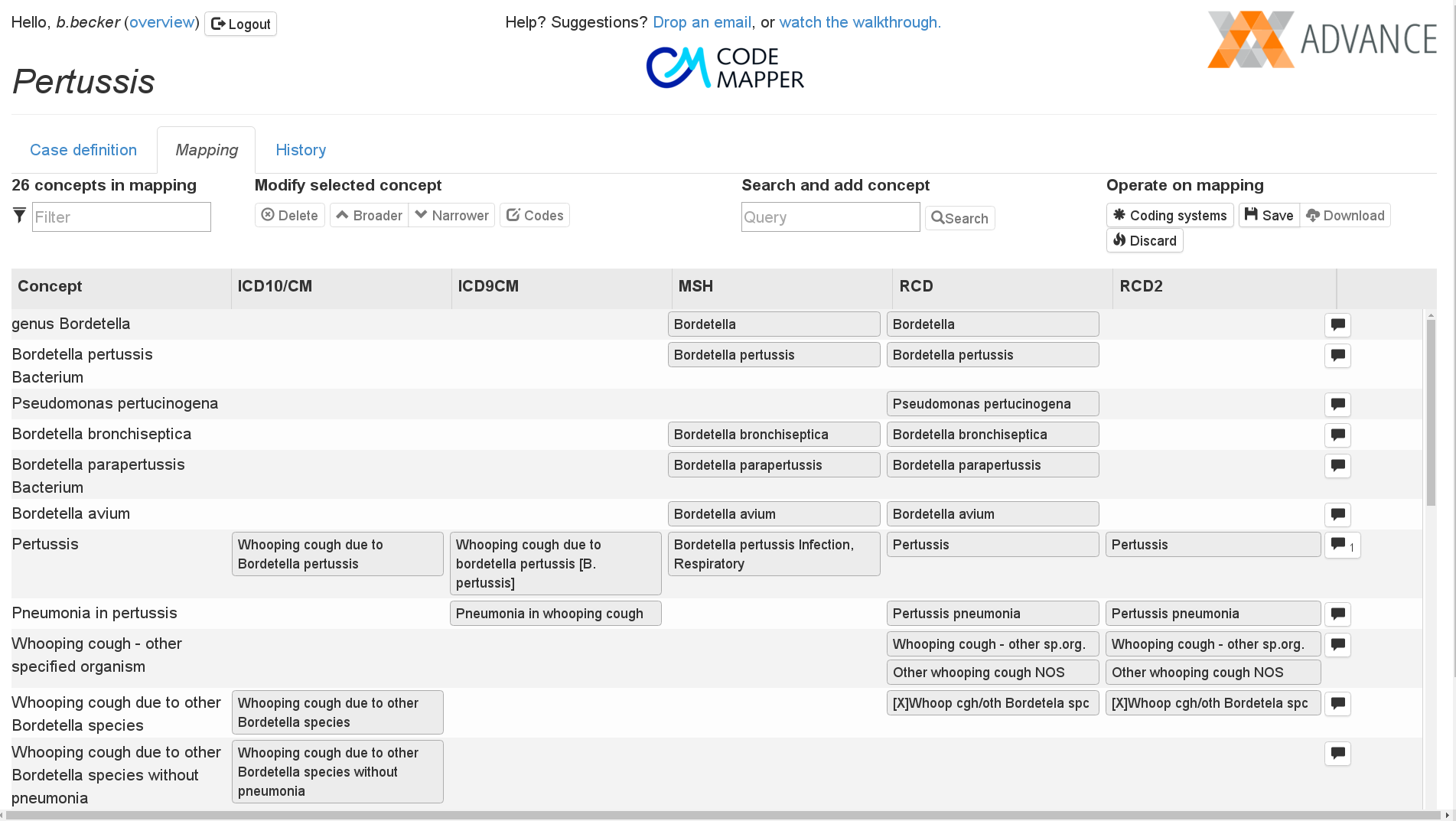


Figure 1: The second screen of the CodeMapper application for revising the mapping

##### Implementation

CodeMapper builds upon information from the Metathesaurus of the Unified Medical Language System (UMLS) [Lindberg1993]. The Metathesaurus is a compendium of various medical coding systems and controlled vocabularies. It provides integration between the coding systems by identifying equivalent codes from different coding systems and assigning them to common concept identifiers. As of version 2015AB, the Metathesaurus maps the codes from 190 coding systems to more than one million medical concept identifiers. The identification of equivalent codes enables aggregating information about medical concepts from different coding systems, such as terms and codes of concepts, and relations to other concepts. Because the identity of medical concepts in CodeMapper are concepts of the Metathesaurus, information from the Metathesaurus can be applied in several processing steps of CodeMapper, which is shown in figure 2 and described in the following.

The automatic identification of medical concepts in the free-text case definition is based on lexical information from the Metathesaurus. The lexical information of a concept contains names and phrases that are used in free-text to refer to that concept (see bottom left of figure 2). Lexical information in the Metathesaurus is aggregated over its entire source coding systems. To identify concepts on the first screen of CodeMapper automatically, the text-indexing engine Peregrine [Schuemie07] is applied together with this lexical information from the Metathesaurus.

Medical concepts are assigned in the UMLS to one or more of 136 semantic types, which define broad conceptual categories like *Disease or syndrome*, *Finding*, or *Substance*. To provide even broader structure, semantic types are in turn combined into 15 semantic groups [McCray2001]. The preselection of concepts after automatic identification by Peregrine in the first screen is determined by the concept’s membership to the semantic group of *Disorders*.

The assignment of equivalent codes from various coding systems to unique medical concepts in the Metathesaurus drives the creation of code sets on the second screen of CodeMapper. Each concept in CodeMapper is projected on all codes that are associated with the concept in the targeted coding systems (see bottom right of figure 2).

Finally, the Metathesaurus provides a taxonomic hierarchy that connects one medical concept with another if it is broader (or narrower) than the other (see bottom centre of figure 2). (The Metathesaurus provides also other relations that are currently not used in CodeMapper.) The Metathesaurus aggregates the hierarchical information that is available in its source coding systems, and additionally defines a distinct hierarchy on top of them. CodeMapper’s revision operations for expanding concepts follow the hierarchical information of the Metathesaurus by retrieving concepts that are connected as narrower (or broader) in the Metathesaurus. Relations from the source coding systems and the distinct hierarchy of the Metathesaurus are taken into account. A lower performance was found in the evaluation when using only subsets of those relations for expansion.

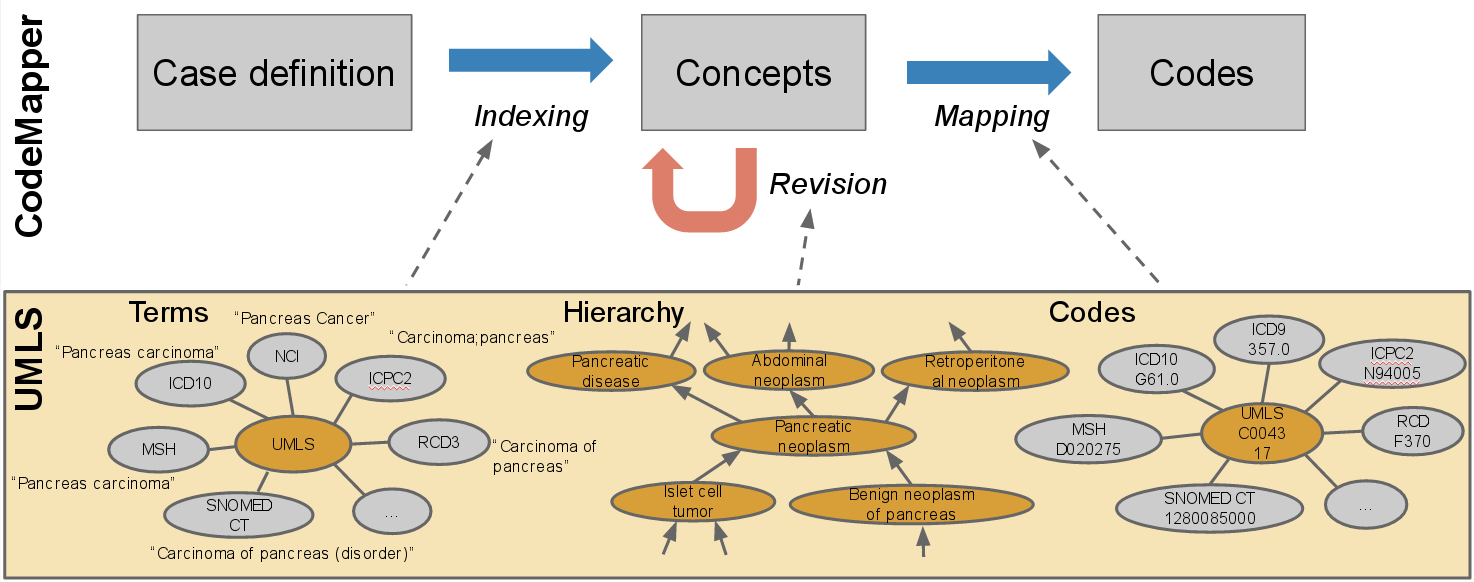


Figure 2: Use of UMLS Metathesaurus in CodeMapper.

##### Evaluation

In the remainder of this article, we present an evaluation of the efficiency of CodeMapper, focussing on the combined steps of automatic concept identification and concept expansion (see figure 3). The evaluation is based on reference code sets for a number of events that have been created manually in a previous project from textual case definition. For each event, we first applied the concept identification step of CodeMapper on the case definition that was also used for the reference sets, and applied then an iterative, automated revision strategy to improve the mapping. The revision strategy comprised the expansion of concepts and the simultaneous pruning of false-positive concepts that do not contain codes of the reference mapping. This automatic strategy was chosen to simulate the manual usage of CodeMapper by a user who uses concept expansion to improve the sensitivity of the mapping and who decides the inclusion of expanded concepts by their relevance for the event. We evaluated the performance of the mapping after each expansion by comparing the codes introduced by the mapping with the reference code sets.

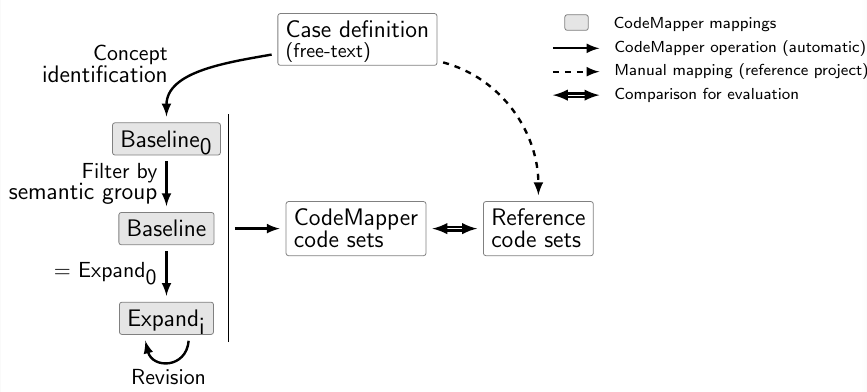


Figure 3: Automatic evaluation of CodeMapper. The 1mappings baseline0, baseline, and expandi on the left are compared with the reference code sets.

The result of the automatic concept identification in the case definition served as the baseline of the evaluation. The mapping *baseline0* contained all concepts that were identified in the case definition. The mapping *baseline* consisted only of concepts that belong to the semantic group of disorders, corresponding to the automatic preselection of concepts on the first screen of the application. The comparison of the mappings *baseline0* and *baseline* provides an evaluation of preselecting concepts by semantic group on the first screen of CodeMapper.

The *baseline* mapping was successively revised by expanding each concept of the mapping to broader or narrower concepts according to the expansion operations in CodeMapper, and retaining expanded concepts only if they contained codes from the reference sets. The mappings *expandi* were created by expanding the concepts successively *i* times.

A mapping was evaluated by comparing the associated code sets of each coding system with the codes of the reference set. This comparison yielded the number of true positive codes (*TP*), false positive codes (*FP*) and false negative codes (*FN*). The sensitivity of a generated code set was defined as *TP/(TP+FN)* and the positive predictive value (PPV) was defined as *TP/(TP+FP)*. For each coding systems the macro-average of sensitivity and PPV over all events in the reference mappings are reported.

##### The *max-sensitivity* mapping

The *expandi* mappings were constructed to improve sensitivity consecutively. To estimate an upper bound of their PPVs we considered the *max-sensitivity* mapping that has maximum sensitivity with maximal PPV. In contrast to *baseline* and *expandi*, the *max-sensitivity* mapping was not derived from the case definition but constructed by first identifying all UMLS concepts that covered at least one code of the reference set and then retaining for each code of the reference set only the concept that introduced the least number of false positives. An example for a *max-sensitivity* mapping is shown in figure 4. The *max-sensitivity* mapping covers all reference codes that are available in the Metathesaurus. The false negatives of *max-sensitivity* are missing in all mapping created with CodeMapper. Every false positive code of *max-sensitivity* issibling of a code of the reference set, where two codes are siblings if they are linked to the same concept in the Metathesaurus. Presuming that a code set is *coherent* between coding systems if the inclusion of one code in a code set implies the inclusion of its siblings, the PPV of *max-sensitivity* serves thus as measure of coherence between coding systems of the reference sets (given consistency of the links between codes and concepts in the Metathesaurus).

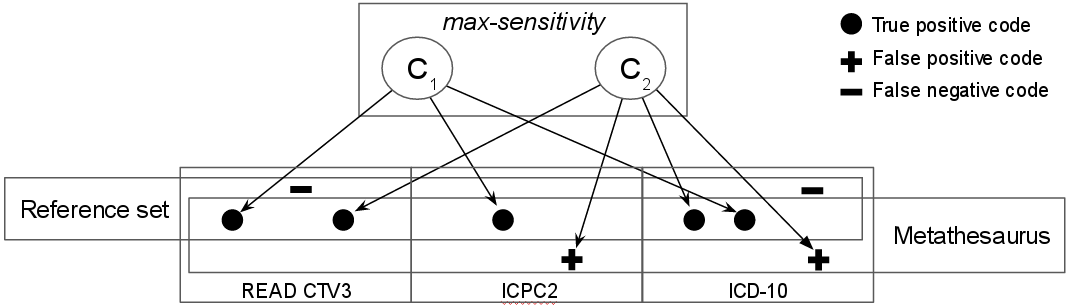


Figure 4: Example of a max-sensitivity mapping that contains two concepts C1 and C2 to cover the codes of the reference set shown as dots below below. False negative codes (**-**) would not be included in any mapping of CodeMapper because they are not in the Metathesaurus. All false positive codes (**+**) are synonyms of true positive codes.

##### Error analysis

We carried out an error analysis of the mapping *expand3* by assigning an error category to every false positive and false negative code. For false positives, we distinguished between codes have siblings in the reference set, and those that have not. By construction of the *expandi* mappings, false positive codes without siblings in the reference mapping were spuriously introduced by the concept identification and not during concept expansion, because only concepts with true positive codes were added in the revisions.

For false negatives, we distinguished between 1) codes that are not assigned in the Metathesaurus and thus cannot be generated by CodeMapper, 2) codes without siblings in the reference set, and 3) codes that would be generated by a further expansion of the mapping.

##### The reference mapping

We used mappings from the SAFEGUARD project[[1]](#footnote-3) [Safeguard2013] of the European Network of Centres for Pharmacoepidemiology & Pharmacovigilance (ENCePP) as a reference for the evaluation. This project was selected for the variety of mapped events, the range of targeted coding systems in use in the European Union and the United States, and the manual mapping process that included review and feedback by database custodians.

The SAFEGUARD project provides textual case definitions for nine events. One event was excluded from the evaluation because the code sets of several target coding systems were missing (sudden cardiac death; the event was captured in only two EHR databases targeted in SAFEGUARD). One event was excluded because the textual case definition was comprised of only a short description of symptoms of the event, which were not directly related to the codes of the mapping (heart failure). The retained events are shown in table 1.

The events were mapped for seven EHR databases with four coding systems: Lombardy regional health records, Medicare, Puglia regional health records, PHARMO (all ICD-9 [ICD9]), GePaRD (ICD-10 [Pavillon1992]), IPCI (ICPC-2 [Hofmans1996]), and CPRD (READ-2 [Schulz1996]). Because different code sets for the same coding system cannot be created with CodeMapper, we selected one reference code set per targeted coding system based on the least number of database-specific additions (ICD-9: Medicare, ICD-10: GePaRD, READ-2: CPRD, ICPC-2: IPCI). Because the reference set of Medicare contained codes from ICD-9 and ICD-9/CM, we integrated a target coding system in CodeMapper that is the union of ICD-9 and ICD-9/CM. The Metathesaurus covers only READ-CTV3 and not READ-2. To generate codes for READ-2 a translation table between READ-2 and READ-CTV3 was integrated into CodeMapper. Codes from the IPCI mapping where trimmed to three digits (e.g. A03) to adjust for the different versions of ICPC that are used in IPCI.

Overall, the reference code sets contains 438 codes for seven events and four target coding systems (table 1). The size of code sets differs between coding systems: The codes sets for READ-2 contain 48.33 codes at average and the code sets for ICPC-2 contain only three codes at average. This discrepancy is firstly due to the differences of granularity of the coding systems (READ-2 has 77290 codes in UMLS and ICPC-2 has 1397). Secondly, the queries to the IPCI database (to which the ICPC-2 code sets are targeted) are supported by keyword searches on the free-text portion of the medical record.

The name and the case definitions of the selected events were provided as an input to CodeMapper. Exclusion codes where not taken into account in the evaluation because they were not specified in the textual case definitions but added based on the database-specific needs.

Table 1: Size of case definitions and number of codes in the reference set. The numbers of exclusion codes are given in brackets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Case definition** | **Codes** | | | |
| **Event** | **(word count)** | **ICD-9** | **ICD-10** | **ICPC-2a)** | **READ-2** |
| Acute Pancreatitis | 49 | 1 (0) | 6 (0) | 1 (0) | 7 (0) |
| Bladder cancer | 87 | 12 (0) | 12 (0) | 1 (3) | 91 (0) |
| Haemorrhagic stroke | 48 | 3 (2) | 22 (2) | 1 (2) | 36 (0) |
| Ischemic stroke | 53 | 10 (0) | 11 (0) | 2 (1) | 20 (0) |
| Myocardial Infarction | 39 | 11 (1) | 7 (0) | 1 (6) | -b) |
| Pancreatic Cancer | 19 | 8 (0) | 9 (0) | 1 (1) | 109 (0) |
| Ventricular Arrhythmia | 234 | 5 (0) | 5 (0) | 1 (1) | 27 (0) |
| Sum | 529 | 50.0 (3.0) | 72.0 (2.0) | 8.0 (14.0) | 290.0 (0.0) |
| Average | 75.57 | 7.14 (0.43) | 10.29 (0.29) | 1.14 (2.0) | 48.33 (0.0) |
|  |  |  |  |  |  |
| a) Additional text-based queries for IPCI database b) Text-based query only for GePaRD database | | | | | |

# Results

##### Baseline

The automatic concept identification in mapping *baseline0* resulted in an average sensitivity of 0.299 for generating the reference code sets (table 2. The sensitivity in each coding system is proportional to the number of codes in the reference mapping. The average PPV of the *baseline0* was0.535. Filtering the concepts by semantic group in mapping *baseline* reduced the number of concepts in all events from 286 to 56. At the same time, the performance changed only slightly (sensitivity of 0.297 and PPV of 0.570) because concepts that did not belong to the semantic group of disease contributed only few codes in the selected coding systems (which in turn focus on diseases).

Table 2: Performance measures of revised mappings in the CodeMapper evaluation.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Revision** |  | **ICD-9** | **ICD-10** | **ICPC-2** | **READ-2** | **Average** |
| baseline0 | Sensitivity | 0.316 | 0.249 | 0.500 | 0.132 | 0.299 |
|  | PPV | 0.486 | 0.606 | 0.700 | 0.347 | 0.535 |
| baseline | Sensitivity | 0.316 | 0.249 | 0.500 | 0.124 | 0.297 |
|  | PPV | 0.514 | 0.606 | 0.700 | 0.458 | 0.570 |
| expand1 | Sensitivity | 0.871 | 0.942 | 1.000 | 0.593 | 0.851 |
|  | PPV | 0.483 | 0.518 | 0.743 | 0.656 | 0.600 |
| expand2 | Sensitivity | 0.900 | 1.000 | 1.000 | 0.787 | 0.922 |
|  | PPV | 0.422 | 0.461 | 0.707 | 0.568 | 0.539 |
| expand3 | Sensitivity | 0.914 | 1.000 | 1.000 | 0.850 | 0.941 |
|  | PPV | 0.426 | 0.455 | 0.707 | 0.550 | 0.534 |
| expand4 | Sensitivity | 0.914 | 1.000 | 1.000 | 0.851 | 0.941 |
|  | PPV | 0.425 | 0.454 | 0.707 | 0.547 | 0.533 |
| max-sensitivity | Sensitivity | 1.000 | 1.000 | 1.000 | 0.930 | 0.982 |
|  | PPV | 0.613 | 0.607 | 0.857 | 0.898 | 0.744 |

#### Concept expansion

The sensitivity of the baseline mapping was improved by expanding the concepts to related concepts in one step (*expand1*, 0.851), two steps (*expand2*, 0.922) and three steps (*expand3*, 0.941). All codes were produced for ICPC-2 in *expand1* and in ICD-10 in *expand2*. The sensitivity increased incrementally for READ-2 and ICD-9. The PPV improved after one expansion step (0.600) and decreased slightly after two (0.539) or three (0.534) expansion steps. A fourth expansion step resulted in only marginal improvement of performance (sensitivity 0.941, PPV 0.533).

##### Max-*sensitivity*

The sensitivity of the *max-sensitivity* mapping was 0.982. All false negative codes belonged to the READ-2 coding system (N=21 in all events, N=14 unique codes). The READ-2 codes were missed out because they were not included in the mapping used to translate from READ-3 codes, or because the translated READ-3 codes were not available in the UMLS. The average PPV of the *max-sensitivity* mapping was 0.744.

##### Error analysis of expand3

False positive codes of *expand3* were generated in all coding systems (N=737, see table 3). Most false positive codes were synonyms of reference codes (N=644; 87.4%). To a smaller extent the false positive codes resulted from false positives from concept identification (N=93; 12.6%). Whereas false positives due to indexing were equally distributed between coding systems, the amount of synonym false positives was highest in the READ-2 code sets and lowest in the ICPC-2 code sets. The differences may partially be explained by the different granularities of the coding systems, which is highest in READ-2 and lowest in ICPC-2.

False negative codes were generated only for READ-2 and ICD-9 (see table 4). False negative codes were mostly READ-2 codes that could not be mapped to UMLS concepts (N=21; 47.7%). (These codes also correspond to the false negatives of *max-sensitivity*.) Other false negative codes were database specific (N=19; 43.2%). Only four reference codes were generated in a fourth expansion step (9.1%).

Table 3: Number of false positive codes in expand3 in error analysis categories and their percentage of all false positive codes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Coding system** | **FP category** | **Count** | **Percentage** |
| ICD-9 | Synonym | 91 | 12.3% |
|  | Indexing | 25 | 3.4% |
| ICD-10 | Synonym | 152 | 20.6% |
|  | Indexing | 47 | 6.4% |
| ICPC-2 | Synonym | 7 | 0.9% |
|  | Indexing | 1 | 0.1% |
| READ-2 | Synonym | 394 | 53.5% |
|  | Indexing | 20 | 2.7% |
| Overall | Synonym | 644 | 87.4% |
|  | Indexing | 93 | 12.6% |

Table 4: Number of false negative codes in expand3 in error analysis categories and their percentage of all false negative codes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Coding system** | **FP category** | **Count** | **Percentage** |
| READ-2 | Not in UMLS | 21 | 47.7% |
|  | DB specific | 13 | 29.5% |
|  | expansion4 | 4 | 9.1% |
| ICD-9 | DB specific | 6 | 13.6% |
| Overall | Not in UMLS | 21 | 47.7% |
|  | DB specific | 19 | 43.2% |
|  | expansion4 | 4 | 9.1% |

# Discussion

In this article, we presented the CodeMapper web application that assists the mapping of textual case definitions to code sets from several coding systems at once, which is often a bottleneck in the implementation of epidemiological multi-database studies. We showed the effectiveness of CodeMapper’s approach by simulating and evaluating an informed usage of the application.

The manual revision of mappings is necessary in CodeMapper but the provided expansion operations are effective for creating realistic code sets. Creating a mapping automatically only by the identification of medical concepts in the case definition was insufficient for reproducing the reference mapping (29.7% sensitivity). Clearly, the mapping process cannot be replaced by a simple indexing step. However, the goal of CodeMapper is to support an informed user in creating such mappings, and CodeMapper’s operations for concept expansion provide an effective and efficient way for this, as we saw in the incremental concept expansion. Most relevant concepts (94.1%) were retrieved by expanding the concepts of the baseline mapping in only three steps.

The manually created reference code sets were not fully coherent between coding systems for two reasons. First, different code sets were created for databases with the same coding systems, which may be necessary to compensate different characteristics of the databases. For example, when an event is underreported in one database, a drug that is usually prescribed in case of the event can be included in the query as a *proxy.* Second, the inclusion of one code did not imply the inclusion of all synonym codes in the reference mapping. The suboptimal PPV of the *max-sensitivity* mapping (74.4%) reflects these incoherencies in the reference mapping. Because every mapping in CodeMapper contains all synonym codes, all false positive codes of *max-sensitivity* are synonym codes missing in the reference mapping. The PPV of *max-sensitivity* constitutes also the maximal PPV that can be obtained by CodeMapper’s approach when optimising sensitivity as we did in the construction of *expandi*. The PPV of the mappings *expandi* should rather be seen in the light of the PPV of the *max-sensitivity* mapping. The PPV of mapping *expand3* reached 71.7% of the PPV of *max-sensitivity*.

Future version of CodeMapper will provide additional retrieval strategies for concepts, for finding concepts that represent proxies of an event, or concepts that are related by codes that co-occur in the databases. The provision of database-specific information (e.g. about code frequencies) in the application will further inform decisions about inclusion and exclusions of codes in the mapping process. CodeMapper could also help the construction of extraction algorithms instead of only the generation of unstructured code sets, for example as database queries to simplify feedback about the code incidences in databases. This can be extended to the processing of structured case definitions (e.g., [Ruggeberg2007,Poli2013]) where generated code sets are compiled to an extraction algorithm according to the logical structure of the case definition.

##### Acknowledgement

We would like to thank the investigators of the SAFEGUARD consortium for the codesets. The research leading to codes has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement № 282521 – the SAFEGUARD project. The CodeMapper application was developed in the ADVANCE project. The research leading to the CodeMapper has received support from the Innovative Medicines Initiative Joint Undertaking under ADVANCE grant agreement № 115557, resources of which are composed of financial contribution from the European Union's Seventh Framework Programme (FP7/2007-2013) and EFPIA companies in kind contribution.

# Literature

[Avillach et al., 2013] Avillach, P., Coloma, P. M., Gini, R., Schuemie, M., Mougin, F., Dufour, J.-C., Mazzaglia, G., Giaquinto, C., Fog/stunari, C., Herings, R., et al. (2013). Harmonization process for the identification of medical events in eight European healthcare databases: the experience from the EU-ADR project. *J Am Med Inform Assoc*, 20(1):184–192.

[Avillach et al., 2009] Avillach, P., Joubert, M., Thiessard, F., Trifirò, G., Dufour, J.-C., Pariente, A., Mougin, F., Polimeni, G., Catania, M. A., Giaquinto, C., et al. (2009). Design and evaluation of a semantic approach for the homogeneous identification of events in eight patient databases: a contribution to the European EU-ADR project. *Studies in health technology and informatics*, 160(Pt 2):1085–1089.

[Coloma et al., 2011] Coloma, P. M., Schuemie, M. J., Trifiro, G., Gini, R., Herings, R., Hippisley-Cox, J., Mazzaglia, G., Giaquinto, C., Corrao, G., Pedersen, L., et al. (2011). Combining electronic healthcare databases in europe to allow for large-scale drug safety monitoring: the eu-adr project. *Pharmacoepidemiology and drug safety*, 20(1):1–11.

[Gini et al., 2016] Gini, R., Schuemie, M., Brown, J., Ryan, P., Vacchi, E., Coppola, M., Cazzola, W., Coloma, P., Berni, R., Diallo, G., Oliveira, J. L., Avillach, P., TrifirÃ², G., Rijnbeek, P., Bellentani, M., van Der Lei, J., Klazinga, N., and Sturkenboom, M. (2016). Data Extraction and Management in Networks of Observational Health Care Databases for Scientific Research: A Comparison among EU-ADR, OMOP, Mini-Sentinel and MATRICE Strategies. *eGEMs (Generating Evidence & Methods to improve patient outcomes)*, 4(1).

[Giuseppe Roberto et al., 2015] Giuseppe Roberto, Ingrid Leal, R. G. Identification of type 2 diabetes cases in a set of databases participating to the EMIF project.

[Hofmans-Okkes and Lamberts, 1996] Hofmans-Okkes, I. and Lamberts, H. (1996). The International Classification of Primary Care (ICPC): new applications in research and computer-based patient records in family practice. *Family Practice*, 13(3):294–302.

[Lindberg et  al., 1993] Lindberg, D. A., Humphreys, B. L., and McCray, A. T. (1993). The Unified Medical Language System. *Methods of information in medicine*, 32(4):281–291.

[Organization, 1975] Organization, W. H. (1975). International Classification of Diseases, 9th Revision, Clinical Modifications.

[Overby et  al., 2013] Overby, C. L., Pathak, J., Gottesman, O., Haerian, K., Perotte, A., Murphy, S., Bruce, K., Johnson, S., Talwalkar, J., Shen, Y., et al. (2013). A collaborative approach to developing an electronic health record phenotyping algorithm for drug-induced liver injury. *Journal of the American Medical Informatics Association*, 20(e2):e243–e252.

[Overhage et  al., 2012] Overhage, J. M., Ryan, P. B., Reich, C. G., Hartzema, A. G., and Stang, P. E. (2012). Validation of a common data model for active safety surveillance research. *J Am Med Inform Assoc*, 19(1):54–60.

[Pavillon and Maguin, 1992] Pavillon, G. and Maguin, P. (1992). The 10th revision of the International Classification of Diseases. *Revue d’epidemiologie et de sante publique*, 41(3):253–255.

[Poli et  al., 2013] Poli, F., Overeem, S., Lammers, G. J., Plazzi, G., Lecendreux, M., Bassetti, C. L., Dauvilliers, Y., Keene, D., Khatami, R., Li, Y., et al. (2013). Narcolepsy as an adverse event following immunization: case definition and guidelines for data collection, analysis and presentation. *Vaccine*, 31(6):994–1007.

[Reich et  al., 2012] Reich, C., Ryan, P. B., Stang, P. E., and Rocca, M. (2012). Evaluation of alternative standardized terminologies for medical conditions within a network of observational healthcare databases. *J Biomed Inform*, 45(4):689–696.

[Rüggeberg et  al., 2007] Rüggeberg, J. U., Gold, M. S., Bayas, J.-M., Blum, M. D., Bonhoeffer, J., Friedlander, S., de Souza Brito, G., Heininger, U., Imoukhuede, B., Khamesipour, A., et al. (2007). Anaphylaxis: case definition and guidelines for data collection, analysis, and presentation of immunization safety data. *Vaccine*, 25(31):5675–5684.

[Schmedt et  al., ] Schmedt, N., Bezemer, I., Berardis, G. D., de Vries, C., Gil, M., Hense, S., Leal, I., Martin, E., Masclee, G., TrifirÃ², G., Requena, G., Rijnbeek, P., Romio, S., Sammon, C., Scotti, L., Seeger, J., Smits, M., Schink, T., Varas-Lorenzo, C., Sturkenboom, M., and Garbe, E. Benchmarking of Definitions and Algorithms to Extract Outcome Events in the EU-funded SAFEGUARD Project. http://www.safeguard-diabetes.org/?q=system/files/private/SAFEGUARD\_DGEpi\_2013\_Benchmarking\_FINAL.pdf. Accessed: 2015-02-12.

[Schuemie et  al., 2007] Schuemie, M. J., Jelier, R., and Kors, J. A. (2007). Peregrine: Lightweight gene name normalization by dictionary lookup. In *Proc of the Second BioCreative Challenge Evaluation Workshop*, pages 131–133.

[Schulz et  al., 1996] Schulz, E., Barrett, J., Brown, P., and Price, C. (1996). The Read Codes: evolving a clinical vocabulary to support the electronic patient record. In *Conference Proceedings: Toward an Electronic Health Record Europe. Newton: CAEHR*, pages 131–40.

[Trifirò et  al., 2014] Trifirò, G., Coloma, P., Rijnbeek, P., Romio, S., Mosseveld, B., Weibel, D., Bonhoeffer, J.,  
Schuemie, M., Lei, J., and Sturkenboom, M. (2014). Combining multiple healthcare databases for postmarketing drug and vaccine safety surveillance: Why and how? *Journal of internal medicine*, 275(6):551–561.

1. http://www.safeguard-diabetes.org/, http://www.encepp.eu/encepp/viewResource.htm?id=8323 [↑](#footnote-ref-3)