Mining clinical big data for drug safety: Detecting inadequate treatment with a DNA sequence alignment algorithm

Thibault Ledieu, Ms[1,2], Guillaume Bouzille, MD[1,2,3], Catherine Plaisant, PhD[4], Frantz Thiessard, MD, PhD[5], Elisabeth Polard, PharmD[3], Marc Cuggia, MD, PhD[1,2,3]
[1]INSERM, UMR 1099, Rennes, F-35000, France;
[2]Université de Rennes 1, LTSI, Rennes, F-35000, France;
[3]CHU Rennes, Rennes, F-35000, France;
[4]Human-Computer Interaction Lab, Univ. of Maryland, College Park MD, USA;
[5]Université de Bordeaux, INSERM, Bordeaux Population Health Research Center, team ERIAS, UMR 1219, F-33000 Bordeaux, France;

Abstract

Health data mining can bring valuable information for drug safety activities. We developed a visual analytics tool to find specific clinical event sequences within the data contained in a clinical data warehouse. To this aim, we adapted the Smith-Waterman DNA sequence alignment algorithm to retrieve clinical event sequences with a temporal pattern from the electronic health records included in a clinical data warehouse. A web interface facilitates interactive query specification and result visualization. We describe the adaptation of the Smith-Waterman algorithm, and the implemented user interface. The evaluation with pharmacovigilance use cases involved the detection of inadequate treatment decisions in patient sequences. The precision and recall results (F-measure = 0.87) suggest that our adaptation of the Smith-Waterman-based algorithm is well-suited for this type of pharmacovigilance activities. The user interface allowed the rapid identification of cases of inadequate treatment.

Introduction

In 1999, an Institute of Medicine publication entitled "*To Err is Human*" highlighted the extent of medical errors[1]. This study reported that between 44,000 and 98,000 deaths were related to medical errors in US hospitals each year. Most of these errors were medication errors[2], frequently caused by negligence[3]. The computerization of hospital information systems has allowed the implementation of computerized prescription systems with appropriate warning messages. These systems are designed to prevent prescription errors[4]. However, they are mainly focused on the detection of harmful drug interactions; therefore, most of them do not take into account other biomedical data to alert physicians. Currently, many research projects reuse and mine electronic health record (EHR) data to evaluate the inconsistency between the physicians' practice and the drug prescription guidelines. EHR data analysis has been facilitated by the emergence of hospital clinical data warehouses (CDW), such as i2b2[5] or eHOP[6], that allow mining large volumes of heterogeneous clinical data.

CDW data are also useful in pharmacovigilance[7,8,9]. Indeed, pharmacovigilance requires reviewing the temporal sequence of the patient's clinical events. This sequence of clinical events (e.g., diagnosis, drug administrations, laboratory tests, signs and symptoms, ward admissions) can be considered as the patient's care trajectory with a beginning and an end. The content of this trajectory will depend on the pharmacovigilance problem under study [10]. In the last years, many visualization and data mining tools have been developed and evaluated for sequence processing. For instance, EventFlow has been used extensively because it allows users to search for exact temporal patterns[11] in a cohort, and to summarize and simplify all sequences in a set of records. Gotz et al. proposed a search methodology[12] based on a query visual interface that allows the user to define a clinical episode the specification of which will be translated into a SQL query to find the patient matching in the database. *Care Pathway Explorer*[13] mines frequent patterns in patient sequence, and then illustrates them in a visually user-friendly interface, but only allows searching for existing patterns in all sequences. However, these tools cannot be used for approximate matches (i.e., fuzzy string search): the retrieved sequences correspond exactly to the query. Similan[14] introduced a similarity metric and a new search interface, but only for short sequences that are pre-aligned to an meaningful date or event. PeerFinder focuses on finding similar individuals based on their attributes and temporal sequences[15,16]. While quite powerful to represent attribute variations in the results, its current implementation is limited to event sequences that have been aggregated (e.g., by week or months) and pre-aligned. To our knowledge, aligning to variants of a search pattern anywhere in the record has not been achieved yet.

To circumvent this limitation, we propose to adapt a DNA sequence alignment algorithm to the pharmacovigilance context. The originality of our approach is that it allows the search and alignment of a reference sequence to a fuzzy event sequence, while taking into account the temporality of events and calculating a similarity score to rank results. Algorithms for DNA sequence alignment (e.g., the Needleman-Wunsch and Smith-Waterman algorithms) have been widely used in bioinformatics to compare and align DNA sequences.[17] The Smith-Waterman (SW) algorithm is one of the most sensitive sequence search algorithms, albeit one of the slowest[18]. Moreover, its usefulness in other fields than bioinformatics, especially in image processing, has been already demonstrated[19]. We started from the hypothesis that finding specific temporal sequences of clinical events/exposure to a drug in a population is similar to finding a defined nucleotide pattern within a DNA sequence.
In this paper, we first describe how we adapted and implemented the SW algorithm in a new sequence visualization visual analytics tool designed for drug safety assessments. Then, we present the early evaluation of this tool with pharmacovigilance use cases to assess its efficiency and efficacy for the detection of inadequate treatments.

Materials and methods

Data source

The data used to implement and evaluate our tool came from eHOP (*entrepôt de données biomédicales de l'HOPital*), the CDW technology we developed at Rennes University Hospital (France)[6]. eHOP integrates all healthcare document types produced by the hospital information system:

- structured data that employ reference terminologies, such as ICD-10 diagnoses from diagnosis-related groups (DRGs), local laboratory test codes, Association for the Development of Informatics in Cytology and Pathology (ADICAP) codes for pathology diagnoses, and Anatomical Therapeutic Chemical (ATC) terminology corresponding to drug prescriptions and administration;
- unstructured data, such as clinical narrative notes, surgical protocols, X-ray or pathology reports.

Some information, such as laboratory results or diagnoses, is recorded both as unstructured and structured data thanks to the corresponding terminological codes. Therefore, eHOP allows users i) to search for information from structured and also unstructured data; and ii) to combine two different ways of data querying. Users can build queries based on reference terminologies, or simply submit keywords to retrieve structured and unstructured documents that contain these terms or keywords. Users can then access documents via a dedicated interface that incorporates functionalities to allow navigating through the entire patient EHR[20]. eHOP is routinely used at Rennes University Hospital to support clinical research.

Currently, the eHOP CDW allows users to search among 40 million unstructured data and 300 million structured elements. All these data are collected from EHRs and cover more than 1.2 million patients.

Model trajectory

We defined our model of patient care trajectory as a chronological ordered list of events. Each event has a character type, a code, a thesaurus, and a time stamp. We only use point events in our model (i.e., only one date). For instance, if the event is denoted E, E_c is the event code, E_t is the event type, E_i is the patient identifier of this event, and E_d is the event date.

Data processing and sequence generation

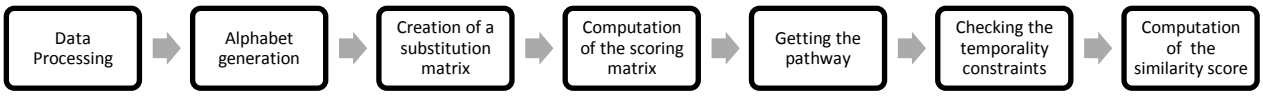

Figure 1. Flowchart of the clinical sequence alignment process.

To fit into the model, we processed the patient's data extracted from the CDW eHOP in the following way. Numerical clinical biology data were discretized according to intervals interactively defined by the user. Drug administration data were processed to identify changes in dosage. In order to find the adverse drug event of interest, diagnoses were extracted from the DRG data using standardized MedDRA Queries to select the ICD-10 codes that corresponded to the diagnosis[21].
The alphabet used for the SW algorithm was a set of pre-defined codes that correspond to the events present in the patient sequence and in the reference sequence, and a code for the null event to represent time periods without any

events. Events were sorted in chronological order to form a sequence. The sequence length is highly variable and can range from a dozen to more than hundred events.

Smith-Waterman algorithm-based tool

Description

We used the Smith-Waterman algorithm, without the Gotoh improvement[22], to compute the optimal local alignment of clinical sequences. The adaptations introduced in the algorithm to fit our specific needs are described in the following sections.

The SW algorithm with its adaptations can be used to compare two sequences: i) the sequence specified by the user (i.e., reference) and ii) a patient' sequence. Each sequence is considered as a string. Each character of this string could represent a clinical event, for instance a drug administration or a laboratory test result. The character types represent the alphabet used by the SW algorithm to compare each sequence. A scoring matrix and a similarity score are computed for each comparison. High scores correspond to higher similarity between the reference sequence and the patient data.

Adaptation to the temporal constraints

As the SW algorithm is not designed to process the temporal information contained in the sequences, we introduced various changes in the algorithm. First, to take into account the time intervals between clinical events, we created a new character type that corresponds to null events for days where there is no event in the patient history.

Second, in some cases, the user is not interested in finding exact matches to a sequence of events, but only the presence of an event before, during or after an alignment of sequences (e.g., "looking for a diagnosis in the days after the specified drug administration pattern and a particular biological measurement"). To solve this problem, we implemented the Allen's interval algebra[23]. Allen's criteria allow us to search for temporal relationships between events. If the desired temporal relationship is not found, the sequence similarity score is lowered.

Computation of the substitution and scoring matrices

We used the substitution matrix (see Figure 1) to calculate the score of each cell in the scoring matrix according to the sequence elements. The substitution matrix was calculated dynamically for each comparison between a patient sequence and the query sequence (defined by the user). If the codes for an event in the two sequences matched, the returned score was the matching score *m*. Otherwise, the cost of the difference was the opposite of the matching score. The returned score *S* was calculated as follows:

$$S(Eci, Ecj) = \begin{Bmatrix} +\, m\ (Eci = Ecj) \\ -\ m\ (Eci \neq Ecj) \end{Bmatrix}$$

where E is the event, c the event code, and i and j the identifiers (reference and patient's sequence, respectively). The scoring matrix allows processing the one-to-one comparisons between all events in the patient sequence and the reference sequence and recording the optimal alignment results. Null events are taken into account during the scoring matrix computation.

The method to calculate the scoring matrix *M* can be summarized as follows:

$$M(i,j) = \ max \begin{Bmatrix} 0 \\ M(i-1, j-1)\ + D(Ai, Bj) \\ M(i-1, j) + \ \Delta \\ M(i, j-1) + \ \Delta \\ M(i-1, j) + \ T(n) \end{Bmatrix}$$

where M is the scoring matrix, D is the function that returns the score in the substitution matrix for the events A and B, Δ the penalty for insertions or deletions, and T is the mathematical function chosen by the user (natural logarithm, exponential…) that calculates the gap penalty for days without event. The value of gap penalty is currently specified in the program, but could be specified by the users, like for Similan[14].

Trace-back process, alignment and checking the temporality constraint

To determine the sequence alignment, we used the scoring matrix to identify the path that could give the maximum alignment score. For this, we assumed that the function D returns the matching score between event A and event B. During the trace-back process, for each i, j (Figure 2):
If M (i, j) = M (i-1, j-1) + D (Ai, Bj), then Ai is matched with Bj and the algorithm goes back to M (i-1, j-1);
If M (i, j) = M (i, j-1) + Δ (gap), then Bj is matched with an empty cell and the algorithm goes back to M (i, j-1);
If M (i, j) = M (i-1, j) + Δ, then Ai is matched with an empty cell and the algorithm goes back to M (i-1, j);
If M (i, j) = M (i-1, j) + T(n), then Ai is matched with "day without event" and the algorithm goes back to M (i-1, j);
If M (i, j) = 0, the local alignment is completed.
This is done for all maximal local M (x, y) to obtain all the optimal local alignments.

X		*INR normal*	*INR too high*	*Null Event*	*INR too high*	*VKA stable*	*VKA rising*	*INR too high*
	0	0	0	0	0	0	0	0
INR too high	0	0	3	3	3	1	0	3
INR too high	0	0	0	0	6	4	2	3
VKA rising	0	0	0	0	4	2	7	5

Figure 2. Example of trace-back process with null events. The reference sequence is in the first column and the patient's sequence in the first row of the matrix. Numbers indicate the matching scores. INR, international normalized ratio; VKA, vitamin K antagonist.

For each temporal constraint, we checked whether the temporally constrained event was located before or after the maximum local alignment score.

Computation of the similarity score

We calculated the similarity score with the following equation:

$$Ssim = \frac{Ml}{Lref * Ms} * \frac{Ntcc}{Ntct}$$

where *Ml* is the maximum local of the scoring matrix, *Lref* the length of the reference sequence, *Ms* the matching score, *Ntcc* the number of checked temporal constraints, and *Ntct* the total number of temporal constraints.
For the final alignment in Figure 3, the similarity score was 77.8%.

INR normal	*INR too high*	*Null Event*	*INR too high*	*VKA stable*	*VKA rising*	*INR too high*
	INR too high	–	*INR too high*	–	*VKA rising*	

Figure 3. Final alignment

Implementation

We wrote the code in Java 8 using the Eclipse Vert.x toolkit (10). Vert.x is an event-driven application framework that uses the JavaVirtual Machine. We used Vert.x to parallelize the computation of the similarity score. We implemented the web interface with the JavaScript libraries JQuery and D3.js.

Interface and visualization

We developed a novel interactive visual interface to query eHOP using the algorithm and visualize the results. Users can build the reference sequence(s) to be searched in a query interface inspired from EventFlow[24] (Figure 4). All event types are listed in the grey area (noted [1] in Figure 4). We developed a visual dictionary for each event type. The discretized numerical events are represented by squares of different colors (e.g., INR measurements are light blue squares). Events that characterize changes of a numerical variable are represented by arrows. The direction of the arrow matches the trend of the change (up, down, stable; e.g., a red arrow is a VKA dosage reduction). Occasional events, such as a diagnosis, are represented by crosses.

Below this area, query sequences can be specified. Users select event types from the dictionary (in the gray area) and drag & drop the icons down to a query line to form a pattern (see area [2] in Figure 4). The spaces to the left and right of the pattern are used to indicate the time constraints (e.g., the [3] area is used to place events that should follow the pattern). In the figure, the specified reference sequence is "four too-high INR measurements (red squares) followed by an increase of VKA dosage (blue arrow)". If needed, multiple patterns can be specified by adding query lines.

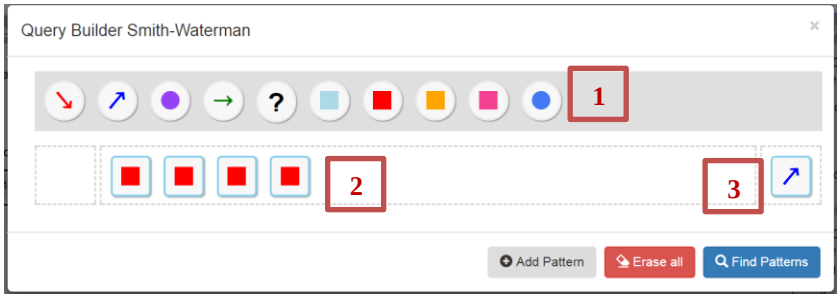

Figure 4. Query Builder based on the Smith-Waterman algorithm. In this example, the user searches for four too-high INR measurements (red squares) followed by an increase of VKA dosage (blue arrow)

The search results are displayed as a list of patients and their sequences, ranked on the basis of their similarity score. In Figure 5, the top six sequences obtained a 100% similarity score, and the three following sequences had a score of 83.33%. Scrolling reveals more sequences. The exact matches are separated from the others by a thin line. Each sequence is aligned to the search pattern or its closest match, shown below the sequence. To narrow the number of sequences to review, users can choose to filter the sequences according to the presence or absence of specific event types ([2] in Figure 5). They can also filter the overall duration of the sequence, or the time between the aligned events and the time-constrained event (i.e., the blue arrow in our example; [3] in Figure 5). All these criteria can be combined.

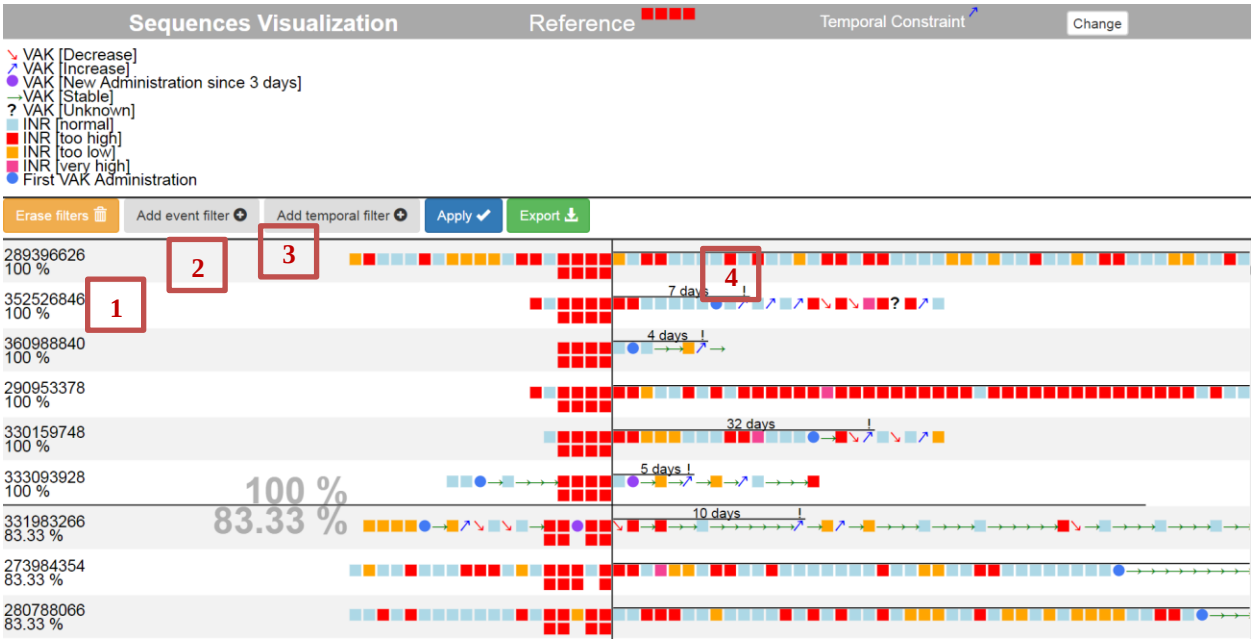

Figure 5. Result visualization. The patient's hospital ID is displayed on the left of the sequence, above the similarity score ([1]). The time interval between the aligned events and the time-constrained event is indicated by a vertical line with on top the time duration in days ([4]).

Evaluation method

To evaluate the relevance of the modified SW algorithm for the identification of clinical event sequences that may correspond to cases of inadequate treatment, we used patients' clinical data extracted from the CDW eHOP. The data extracted for the use case were completely anonymized.

For the use case, we first selected the 10,882 hospital stays that included concomitantly these two event types: 1) international normalized ratio measurements (i.e., INR, a laboratory measurement used to determine the effects of oral anticoagulants) and 2) vitamin K antagonist (VKA) administrations. The aim was to find cases of inappropriate VKA administration. VKA dosage should be reduced when INR is higher than the target value (>3). For this, we searched for administration patterns showing obvious problems, such as an increase - rather than decrease - of the

administered dose after two consecutive INR values above the target value. For such query, the reference event sequence is "INR too high - INR too high - VKA dose increase", and the algorithm produces a score between 0 and 100 for each hospital stay.

We randomly selected 80 sequences that were equally distributed in four similarity score classes ([100;75[, [75;50[, [50,25[, [25,0]).

In the first phase of the evaluation the gold standard was defined by asking a pharmacovigilance expert to review the 80 sequences and to identify all true cases of inadequate treatment, based on the sequence of events displayed on the interface (see next section). To avoid biasing the expert's judgment the sequences were sorted randomly, the similarity score was hidden, and no alignment was done. This gold standard was used to evaluate the performance of the algorithm. We estimated the sensitivity, specificity and F-measure between the gold standard review results and the similarity scores calculated by the SW algorithm. The sample size of 80 sequences allowed us to have a statistical power of 90% with a type I error of 5% to detect an Area Under the Curve (AUC) of at least 0.70 between our gold standard and the classification by the algorithm.

In the second phase of the evaluation, we compared the performance of users reviewing the 80 sequences with and without the help of the algorithm embedded in the tool user interface.

To estimate the time savings allowed by the algorithm ranking system in the search of inadequate treatments, two other expert users reviewed independently the extracted sequences to classify them as corresponding, or not, to cases of inadequate treatment. Like with the first expert, they had to identify inadequate treatments based on the events in these 80 sequences. They repeated the task twice with two different methods, with a break of one day between methods to avoid a learning effect:

1. Random order: they first reviewed the sequences without the help of the algorithm. Sequences were listed in random order and no similarity score was provided. No sequence alignment was performed.
2. Sorted by similarity: the second time they had the help of the algorithm. Sequences were aligned, and sorted according to their similarity to the reference sequence. Users were told that the most similar sequences were at the top, but the similarity score was not provided.

The two experts reviewed and analyzed the sequences one by one, and then reported in a file: the sequence number, whether it was a case of inadequate treatment or not, and the similarity score (for method 2). We recorded the time required to analyze the 80 sequences, for each user and each method. We computed the average time needed to review a sequence with each method, and compared them using the Wilcoxon paired test. Statistical analyses were performed with the R statistical software, version 3.4.1.[25]

Evaluation Results

Algorithm performance

The results of the pharmacovigilance expert (gold standard review) were used to evaluate the performance of the SW algorithm for the identification of sequences of inappropriate drug administration cases. The accuracy of the algorithm was 1 (i.e., all sequences with a similarity score of 100% concerned inadequate treatments), the recall was 0.76, the F-measure was 0.866, and the AUC was 0.99 (Receiver Operating Characteristic curve in Figure 6).

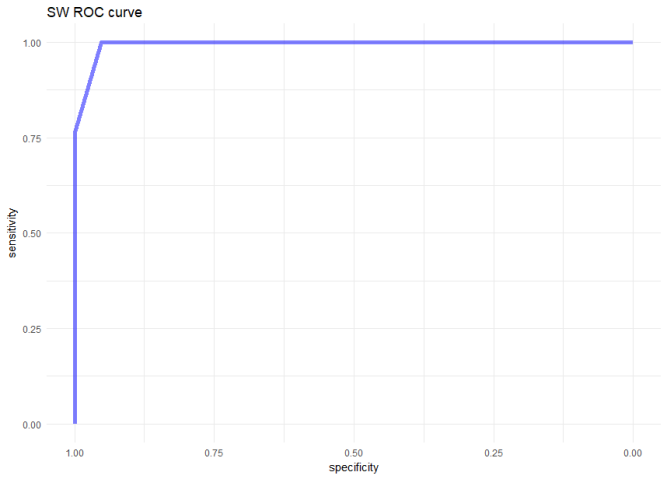

Figure 6. Receiver Operating Characteristic (ROC) curve showing the classification ability of the adapted Smith-Waterman algorithm.

Time of analysis

For user #1 the average analysis time of a sequence was 2.26 ± 2.86 seconds with sorted and aligned sequences and 3.49 ± 3.54 seconds for sequences in random order ($p = 0.0003$, Wilcoxon test) (Figure 7 - left). For user #2, the average time was 1.06 ± 1.06 seconds with ordered and aligned sequences and 1.53 ± 0.85 seconds for unordered sequences ($p < 0.0001$, Wilcoxon test).

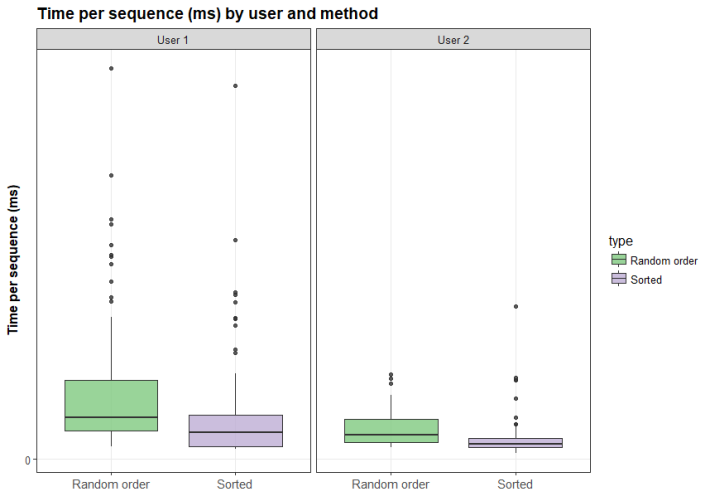

Figure 7. Boxplot showing the distribution of time spent per sequence, for each user and method.

Discussion

The modified SW algorithm allows finding sequences of interest in the patient files represented in the form of sequences. This approach has the advantage of allowing fuzzy string searching. The modifications we introduced in the SW algorithm takes into account the temporal information in the sequences, allow the alignment of the sequence according to the chosen pattern, thus overcoming the limitations of previous tools - for example PeerFinder that only aligns the sequences relative to the date of the first event.

The evaluation results demonstrate the relevance of the SW algorithm for the detection of inappropriate drug administrations. In our use cases, the ranking system allowed a quick and precise identification of the patterns of interest in patient sequences. In addition, sequences that do not present the exact pattern of interest may also be relevant. For example, in the use cases, the variant pattern "INR too high" - "INR too high" - "VKA dose administered stable - "VKA dose administered increased", also indicated an inappropriate drug administration

because the administered dose was not reduced, according to the current recommendations. This is an illustration of the relevance of fuzzy pattern search with the adapted SW algorithm. The AUC value (0.99) indicates that the SW algorithm is a very good classifier for this use case.

The time evaluation did not include the time to specify the query, or the time to run the algorithm. These will vary depending on the size of the dataset and the query. We ran the program with a growing number of sequences and several reference sequence lengths. These tests were carried out with a machine with an Intel Xeon CPU E5-2603 v3 (1.90 GHz). For a reference sequence of length 3, the processing time is 2 sec. and 154 ms for 100 sequences, 19 sec. and 358 ms for 1000 sequences, 3 min., 8 sec. and 583 ms for 10,000 sequences. For a reference sequence of length 12, the processing time is 2 sec. and 301 ms for 100 sequences, 21 sec. and 35 ms for 1000 sequences, 3 min., 21 sec. and 381 ms for 10,000 sequences. The computation time required for the algorithm is linear depending on the number of sequences, while the length of the sequence has little effect. In summary the use of the algorithm can delay the start of the human review of sequences, but we believe that it is likely to accelerate the review itself, especially if few similar sequences are found.

The SW algorithm has limitations: it is inadequate for searches of distant patterns in time. For example, medication for hyperthyroidism for several weeks followed by hospitalization for heart problems is a pattern that will not be found with our tool. By design, the original SW algorithm allows retrieving only near-term event patterns. Other tools (e.g., EventFlow) can search for patterns with distant events, but only finds exact matches.
Moreover, the modifications made to the SW algorithm do not allow searching sequence patterns in which a specific event is absent (for instance, absence of prophylaxis before surgery).This limitation can be overcome in simple cases by searching first for all sequences that include the opposite event (in our example, "prophylaxis before surgery") and by removing all the matching records before starting a new search.
In our case study, we found that the CDW data pre-processing step was essential for the sequence search (e.g., by finding changes in dosage). The sequences, and thus the patterns visibility, will vary depending on the choices made by the user during pre-processing. The input of medical experts is essential to generate sequences suitable for the research question[26].

To improve the identification of similar event sequences, we plan to cluster sequences. Specifically, by considering patterns as terms and all sequences as a corpus, we can calculate a term frequency-inverse document frequency (tf-idf) for each pattern in each sequence. This gives a vector of length m, where n is the number of sequences in the corpus, and m the number of sub-sequences in the corpus. Patterns can then be extracted using an algorithm, such as the Generalized Sequential Pattern algorithm, and vectors clustered using a K-means algorithm.

Conclusion

The adapted SW algorithm allows finding sequences of interest in the patient EHR represented in the form of sequences. This approach has the advantage of allowing fuzzy string searching, and the modifications introduced in the SW algorithm allow taking into account the temporal dimension in sequence processing.

Acknowledgements

This work was supported by the Pharmaco-Epidemiology of Health Products (PEPS) consortium. We also thank Dr. Elisabetta Andermarcher for her careful proofreading of the paper.

References

1. Institute of Medicine (US) Committee on Quality of Health Care in America. To Err is Human: Building a Safer Health System [Internet]. Kohn LT, Corrigan JM, Donaldson MS, editors. Washington (DC): National Academies Press (US); 2000 [cited 2018 Jan 23]. Available from: http://www.ncbi.nlm.nih.gov/books/NBK225182/
2. Leape LL, Brennan TA, Laird N, Lawthers AG, Localio AR, Barnes BA, et al. The nature of adverse events in hospitalized patients. Results of the Harvard Medical Practice Study II. N Engl J Med. 1991 Feb 7;324(6):377–84.
3. Brennan TA, Leape LL, Laird NM, Hebert L, Localio AR, Lawthers AG, et al., Harvard Medical Practice Study I. Incidence of adverse events and negligence in hospitalized patients: results of the Harvard Medical Practice Study I. 1991. Qual Saf Health Care. 2004 Apr;13(2):145–51; discussion 151–2.

4. Bates DW, Leape LL, Cullen DJ, Laird N, Petersen LA, Teich JM, et al. Effect of computerized physician order entry and a team intervention on prevention of serious medication errors. JAMA. 1998 Oct 21;280(15):1311–6.
5. 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). J Am Med Inform Assoc. 2010 Jan 3;17(2):124–30.
6. 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;169:584–8.
7. Osmont M-N, Campillo-Gimenez B, Metayer L, Jantzem H, Rochefort-Morel C, Cuggia M, et al. [Peranesthesic Anaphylactic Shocks: Contribution of a Clinical Data Warehouse]. Therapie. 2015 Oct 16;
8. Beuscart R. PSIP: an overview of the results and clinical implications. Stud Health Technol Inform. 2011;166:3–12.
9. Wang W, Kreimeyer K, Woo EJ, Ball R, Foster M, Pandey A, et al. A new algorithmic approach for the extraction of temporal associations from clinical narratives with an application to medical product safety surveillance reports. J Biomed Inform. 2016 Aug;62:78–89.
10. Le Meur N, Gao F, Bayat S. Mining care trajectories using health administrative information systems: the use of state sequence analysis to assess disparities in prenatal care consumption. BMC Health Serv Res [Internet]. 2015 May 15 [cited 2015 Oct 14];15. Available from: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4436876/
11. Monroe M, Rongilan L, Juan Morales del Olmo, Shneiderman B, Plaisant C, Millstein J. The challenges of specifying intervals and absences in temporal queries: A graphical language approach. In ACM; 2013. p. 2349–58.
12. Gotz D, Wang F, Perer A. A methodology for interactive mining and visual analysis of clinical event patterns using electronic health record data. J Biomed Inform. 2014 Apr 1;48:148–59.
13. Perer A, Wang F, Hu J. Mining and exploring care pathways from electronic medical records with visual analytics. J Biomed Inform. 2015 Aug 1;56:369–78.
14. Wongsuphasawat K, Shneiderman B. Finding Comparable Temporal Categorical Records: A Similarity Measure with an Interactive Visualization. In 2009.
15. Du F, Plaisant C, Spring N, Shneiderman B. Finding Similar People to Guide Life Choices: Challenge, Design, and Evaluation. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems [Internet]. New York, NY, USA: ACM; 2017 [cited 2018 Jan 23]. p. 5498–544. (CHI '17). Available from: http://doi.acm.org/10.1145/3025453.3025777
16. Du F, Plaisant C, Spring N. Visual Interfaces for Recommendation Systems: Finding Similar and Dissimilar Peers. In 2018.
17. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol. 1981 Mar 25;147(1):195–7.
18. Huang L-T, Wu C-C, Lai L-F, Li Y-J. Improving the Mapping of Smith-Waterman Sequence Database Searches onto CUDA-Enabled GPUs. BioMed Res Int [Internet]. 2015 [cited 2018 Feb 14];2015. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4538332/
19. Dumont E, Merialdo B. Rushes Video Parsing Using Video Sequence Alignment. In: 2009 Seventh International Workshop on Content-Based Multimedia Indexing. 2009. p. 44–9.
20. Ledieu T, Van Hille P, Bouzille G, Renault E, Cuggia, Marc. Semantic and Interactive Timeline for Patient Data Visualization [Internet]. 2015 [cited 2018 Jun 18]. Available from: https://knowledge.amia.org/59310-amia-1.2741865/t005-1.2744350/f005-1.2744351/2249164-1.2744998/2248496-1.2744995?qr=1
21. Standardised MedDRA Queries | MedDRA [Internet]. [cited 2018 Feb 14]. Available from: https://www.meddra.org/standardised-meddra-queries
22. Gotoh O. An improved algorithm for matching biological sequences. J Mol Biol. 1982 Dec 15;162(3):705–8.
23. Allen JF. Maintaining Knowledge About Temporal Intervals. Commun ACM. 1983 Nov;26(11):832–43.
24. Monroe M, Lan R, Lee H, Plaisant C, Shneiderman B. Temporal Event Sequence Simplification. IEEE Trans Vis Comput Graph. 2013 Dec;19(12):2227–36.
25. R Development Core Team (2008). R: A language and environment for statistical computing [Internet]. R Foundation for Statistical Computing. Vienna, Austria; Available from: https://www.r-project.org/
26. Fan Du null, Shneiderman B, Plaisant C, Malik S, Perer A. Coping with Volume and Variety in Temporal Event Sequences: Strategies for Sharpening Analytic Focus. IEEE Trans Vis Comput Graph. 2017 Jun;23(6):1636–49.