Learning to Identify Inappropriate Antimicrobial Prescriptions

Mathieu Beaudoin[1], Froduald Kabanza[1], Vincent Nault[2], and Louis Valiquette[2]

[1] Dept. of Computer Science, Université de Sherbrooke, Canada
{mathieu.beaudoin,froduald.kabanza}@usherbrooke.ca
[2] Dept. of Microbiology and Infectiology, Université de Sherbrooke, Canada
{vincent.nault,louis.valiquette}@usherbrooke.ca

Abstract. Inappropriate antimicrobial prescribing is a major clinical problem and health concern. Several hospitals rely on automated surveillance to achieve hospital-wide antimicrobial optimization. The main challenge in implementing these systems lies in acquiring and updating their knowledge. In this paper, we discuss a surveillance system which can acquire new rules and improve its knowledge base. Our system uses an algorithm based on instance-based learning and rule induction to discover rules for inappropriate prescriptions. The algorithm uses temporal abstraction to extract a meaningful time interval representation from raw clinical data, and applies nearest neighbor classification with a distance function on both temporal and non-temporal parameters. The algorithm is able to discover new rules for early switch from intravenous to oral antimicrobial therapy from real clinical data.

Keywords: Classification, temporal data mining, interval sequence, instance-based learning, nearest-neighbor, antimicrobial optimization.

1 Introduction

Inappropriate antimicrobial (ATM) prescribing is a major clinical problem and health concern, with as many as 50% of ATM prescriptions being unnecessary or inappropriate [1]. ATM stewardship programs have been shown to reduce avoidable adverse effects (toxicity, ATM resistance, *Clostridium difficile*, etc. [1,2]) and length of stay, improve patient health, and reduce unnecessary costs. ATM optimization requires the revision of an overwhelming amount of clinical data by dedicated experts, which proves to be an obstacle in the current context of limited healthcare resources. Therefore, several hospitals rely on automated decision support systems to revise hospital-wide ATM prescriptions.

Over the past five years, we have implemented, deployed, and evaluated an automated system called APSS – antimicrobial prescription surveillance system. It is currently deployed at the *Centre Hospitalier Universitaire de Sherbrooke* (CHUS), a Canadian academic centre of 713 beds. It uses expert rules to identify mismatches between prescribed ATMs and published and local guidelines.

N. Peek, R. Marín Morales, and M. Peleg (Eds.): AIME 2013, LNAI 7885, pp. 248–257, 2013.

A clinical pharmacist first reviews the documented alerts and then contacts the prescribing physician to recommend a prescription modification or discontinuation if deemed appropriate. Over the last two years, pharmacists have rejected as many as 50% of false alerts. On the other hand, 91% of the alerts retained (3 156 total) were accepted by the prescribing physicians. This has contributed to decrease intravenous ATM consumption by 22% and ATM expenses by 688 000 CAD. APSS enabled us to extend our surveillance from high-risk wards (e.g., intensive-care) to every bed of the CHUS' two physical sites.

The high proportion of false alerts generated by APSS is mainly explained by a number of prescription parameters that require fine-tuning, changes in prescription guidelines that are not accurately updated, and other factors that affect the appropriateness of a prescription that are not explicitly accounted for in the guidelines and hence not encoded in the knowledge base. The pharmacists' revision process is impeded by this high rate of false alerts.

In order to reduce the proportion of false alerts generated by APSS, we have been investigating the use of a machine learning algorithm that discovers new rules for classifying inappropriate prescriptions, supervised by user feedback such as the rejection of false alerts by the pharmacist or physician, or the identification of unflagged inappropriate prescriptions. The objective is to automatically improve the knowledge base of APSS based on experience. Given that prescriptions are temporal data by nature, we use a supervised learning algorithm for discovering rules that classify temporal data – this is a binary classification into good and bad temporal data (i.e., prescriptions). The algorithm we use is a combination of rule induction and instance-based learning methods. The application of machine learning to clinical temporal data is not new. We review some applications below. However, to the best of our knowledge this is the first application to the monitoring of ATM prescribing.

To illustrate the temporal nature of ATM prescribing, consider this example. A physician chooses a treatment after the first assessment of a patient. As new information becomes available, he will modify the initial treatment to account for clinical and laboratory test results and variations in the patient's state of health. A key intervention in ATM prescribing is *early switch therapy* where an intravenous ATM is replaced by an oral ATM providing a less costly alternative and allowing the patient to be discharged earlier. An early switch typically occurs after 72 hours of intravenous ATM treatment, if the patient is able to take oral medications and his condition has been stable over the last 48 hours.

In the rest of this paper, we first give an overview of related work. We then describe the supervised learning algorithm that we have integrated into APSS and discuss preliminary experimental results. We conclude with future work.

2 Related Work

There are various applications of data mining and machine learning algorithms to clinical temporal data, including temporal abstraction which is commonly used to extract a meaningful representation of the raw data using qualitative time

intervals [3]. Association rule discovery has been used to gain insight into causes of clinical events of interest (e.g.,[4,5]); however it is geared towards discovering rules for frequent patterns and performs poorly when addressing infrequent patterns [6] such as inappropriate prescriptions. It uses an *Apriori*-like strategy [7] with breadth-first search and candidate pruning based on *support* and *confidence*. A problem with this strategy when looking at infrequent patterns is the necessity to lower support thresholds. It inefficiently prunes the candidate space and potentially leads to an intractable search space. Furthermore, it produces an overwhelming quantity of uninteresting patterns from which it is difficult to distinguish interesting ones [6].

Another method used to identify clinical events of interest is case-based reasoning. For example, case-based reasoning has been used to identify potential adverse drug events [8] and hemodialysis treatment failures [9] by looking for similar past cases. While case-based reasoning and instance-based learning are known to perform well with few instances, they are burdened with irrelevant attributes [10] and accumulate large quantities of cases. This is a problem when looking for a small set of highly accurate, concise and intelligible rules aimed at a human user.

A complementary approach to instance-based learning is rule induction. Rule induction is known for its ability to dispose easily of irrelevant features, separate classes with good accuracy, and extract a small set of rules that can lead to better predictions [10]. However, it tends to be affected by a skewed distribution of classes and produce rules that favor the overrepresented classes [11]. Combining instance-based learning and rule induction has been known to address their respective limits with their complementary strengths with traditional non-temporal feature-value data [10].

Our machine learning algorithm also combines instance-based learning and rule induction. However, unlike the approach in [10], which learns classification rules for a labeled set of non-temporal feature-value data, our algorithm learns classification rules for a labeled set of qualitative time interval sequences in addition to non-temporal feature-value data. Before describing the algorithm, we first explain the context of application more precisely and state the problem solved by our algorithm more formally.

3 Application Context and Formal Problem Statement

APSS communicates with the CHUS' electronic health record system and receives administrative and clinical data for every adult inpatient under ATM therapy. For the experiments discussed later in this paper, we selected every adult patient admitted between January 1st 2012 and June 30th 2012. We considered the following attributes: *gender*, *age*, Body Mass Index (*BMI*), patient location (*ward*), temperature (*temp*), white cell count (*WCC*), neutrophil count (*neut*), creatinine clearance (*CrCl*), respiratory rate (*resp*), *pulse*, and blood pressure (*BP*). An attribute was also created for each medication. Prescriptions were described using their *name*, *dose*, *frequency*, and *route* of administration.

We pre-processed this heterogeneous data using simple temporal abstraction mechanisms to extract a uniform and meaningful representation. Figure 1 illustrates the process of state abstractions for the raw *temp* time series where quantitative thresholds were used to identify qualitative states, which we call *episodes*, that hold over a period of time. A temporal granularity of 1 hour was defined. We extracted a single sequence for each hospitalization. Our observation period was restricted to the ongoing ATM of interest, where we considered only data between the first ($t_{\min}$) and last ($t_{\max}$) administered dose. It ensures a common time zero ($t_{\min}$) between sequences.

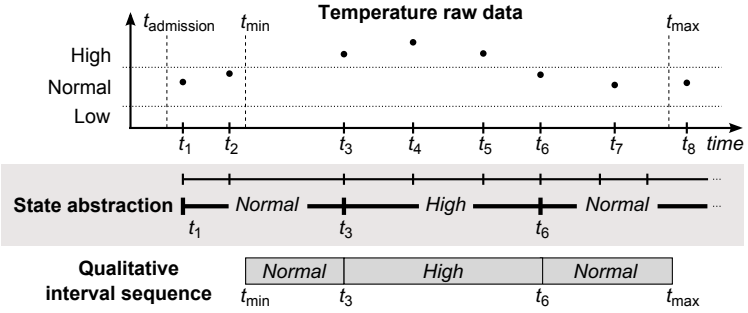

Fig. 1. Example of state abstractions for the *temp* attribute

Let us consider the attribute space A as the finite set of attributes for our domain and the feature space F as the finite set of qualitative states observed for these attributes. An *episode* e is defined as $< a, f, ts, te >$, where $(a = f)$ describes a symbolic state with $a \in A$ and $f \in F$ holding over the time interval $[ts, te[$. We refer to the attribute, feature, start, and end times of an episode as $e.a$, $e.f$, $e.ts$, and $e.te$ respectively. An example of episode from Fig. 1 is $< temp, normal, t_{\min}, t_3 >$.

A *sequence* s is defined by $\{e_1, \ldots, e_n | \forall i = 1, \ldots, n-1 : e_i.ts \leq e_{i+1}.ts\}$, where $n = |s|$, the size of the sequence. We refer to the subsequence of s for the ith attribute $a_i \in A$ as $att_i(s)$ defined by $\{e_1, \ldots, e_m | \forall e \in att_i(s) : e \in s; e.a = a_i; \forall j = 1, \ldots, m-1 : e_j.te \leq e_{j+1}.ts\}$, where $m = |att_i(s)|$. A hospitalization is described as a *labeled sequence* ls defined as $\{id, s, l\}$, where id is a unique identifier, s is a sequence, and l is a class label that belongs to the finite set of class labels L. We focus on a binary-class problem where $L = \{negative, positive\}$. We used APSS' revised alerts to label every sequence, where *positive* indicates a true positive and *negative* indicates a negative or false positive.

We can now formally state the supervised machine learning problem that concerns us. Given the finite training set TS of labeled sequences, discover a rule set R for classifying positive sequences. We only have two classes (positive and negative). Learned classification rules identify positive instances. The antecedent of a learned rule is a conjunction of propositions over time intervals whose satisfaction implies membership to the *positive* class; the consequent is *true*.

4 Temporal Induction of Classification Models

Our supervised learning algorithm, called *Temporal Induction of classification Models* (TIM) combines instance-based learning and rule induction. Its main operations are the following: at first, the rule set R is initialized using positive sequences of the training set as maximally specific rules. Distances between rules and sequences of the training set are computed and stored in a multidimensional distance matrix to reduce computation times. These distances are used for nearest neighbor classification. Rules are modified in parallel to increase interclass distance. At each iteration, the most promising local modifications are selected according to the rule's most similar negative sequences. Conditions are eliminated or their time intervals are shortened. Local modifications are performed according to similar negative sequences until they no longer improve a rule.

The rules are evaluated according to the *J-measure* [12], which quantifies the average information content of a rule. We selected the *J-measure* for its ability to account for both simplicity and *goodness-of-fit*, measuring the probability and *cross-entropy* of a rule [12]. As a working hypothesis, a rule with high information content (i.e., high probability and cross-entropy) is also likely to have a high predictive accuracy.

4.1 Classification

The distance function measures the similarity between rules and sequences, where rules classify sequences that involve temporal and non-temporal data. Accordingly, we use a distance function that considers both temporal and non-temporal parameters. A non-symmetric distance function is used where similarity is proportional to the number of conditions that a sequence shares with a rule, i.e., a sequence is perfectly similar to a rule it subsumes.

Given a rule $r \in R$ with N_r attributes and a sequence $s \in TS$, the global $distance(r, s)$ function is defined by Equation (1). Normalizing $distance(r, s)$ by N_r creates a coefficient between $[0, 1]$, where 0 denotes perfect similarity, that does not arbitrarily favor shorter rules. To ensure that irrelevant sequences are not labeled as positive by the nearest, yet dissimilar, rule we enforce a minimal distance threshold $D_{\min}$ under which a rule is said to **cover** a sequence. In such case, the sequence is labeled as positive by the rule.

$$distance(r, s) = \frac{\sum_{i=1}^{N_\mathrm{r}} D_\mathrm{a}(att_i(r), att_i(s))}{N_\mathrm{r}} \tag{1}$$

The D_a function measures the distance between subsequences $att_i(r)$ and $att_i(s)$ for the ith attribute of r. If $att_i(s) = null$, $D_\mathrm{a} = 1$, otherwise we use Equation (2) which measures the distance between the conditions $c_j \in att_i(r)$ and episodes $e_k \in att_i(s)$. An indexing mechanism retrieves attribute-specific subsequences in O(1). We normalize the distance $D_\mathrm{a} \in [0, 1]$ to avoid arbitrarily increasing the weight of the ith attribute in the global coefficient.

$$D_{\mathrm{a}}(att_i(r), att_i(s)) = \frac{|att_i(r)| - \left(\sum_{j=1}^{|att_i(r)|} \sum_{k=1}^{|att_i(s)|} (S_{\mathrm{F}}(c_j, e_k) \times S_{\mathrm{T}}(c_j, e_k))\right)}{|att_i(r)|} \tag{2}$$

Feature Similarity. The feature similarity function S_{F} measures the similarity between the symbolic features of c_j and e_k using the *overlap metric* where $S_{\mathrm{F}}(c_j, e_k) = 1$ if $(c_j.f = e_k.f)$ and 0 otherwise.

Temporal Similarity. Temporal similarity is proportional to the temporal overlapping of e_k over c_j, as measured by Equation (3). S_{T} returns a coefficient between $[0, 1]$, where 1 implies $[c_j.ts, c_j.te[\subseteq [e_k.ts, e_k.te[$.

$$S_{\mathrm{T}}(c_j, e_k) = \frac{[c_j.ts, c_j.te[\cap [e_k.ts, e_k.te[}{[c_j.ts, c_j.te[} \tag{3}$$

Consider the attribute-specific subsequences of Fig. 2. A rule's antecedent $att_i(r)$ with conditions c_1 and c_2 overlaps a sequence's $att_i(s)$ with episodes e_1, e_2 and e_3. The distance between these subsequences is 0.2, which is computed as follows:

$$\begin{aligned} D_{\mathrm{a}}(att_i(r), att_i(s)) &= \frac{2 - \left(\sum_{j=1}^{2} \sum_{k=1}^{3} (S_{\mathrm{F}}(c_j, e_k) \times S_{\mathrm{T}}(c_j, e_k))\right)}{2} \\ &= \frac{2 - ((1 \times 0.6) + (0 \times 0.4) + (1 \times 0) + (0 \times 0) + (1 \times 1) + (0 \times 0))}{2} \\ &= 0.2 \end{aligned}$$

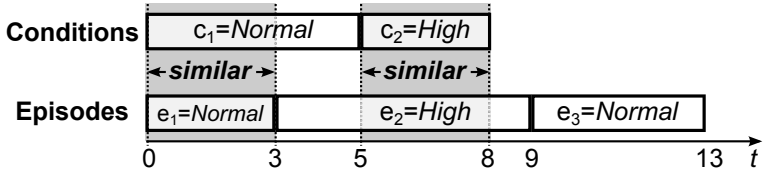

Fig. 2. Example of a rule's conditions and a sequence's episodes

4.2 Refinement of the Rule Set

The intuition behind this rule refinement process is that increasing interclass distance creates more accurate rules. Rules are modified in parallel, where each iteration provides a set of locally promising modifications. Promising modifications are selected by comparing a rule and its most similar negative sequence. Rules are modified by removing the temporal overlapping between a condition

c and an episode e, resulting in a modified condition c' being either entirely removed or subsumed by c.

Since a new rule is subsumed by the original, it will only be similar to sequences that were already (partially) similar to the original rule. Thus, distances must only be updated for these sequences. TIM uses a multidimensional *distance matrix* to keep track of the partially similar sequences of a rule.

5 Results

As a preliminary experiment for our algorithm, we have tested TIM with learning rules that identify"*early switch therapy*". A clinically valid recommendation for early switch from intravenous to oral ATM therapy requires the following three indications: 72 consecutive hours of intravenous therapy, 48 hours of stabilized state of health (e.g., normal levels of white cell count and temperature), and 24 hours of concurrent oral therapy. This rule involves non trivial temporal constraints, making it a good test case for the learning algorithm. In this experiment, this rule is not specified. The dataset only contains positive and negative labels specifying if a hospitalization contains or not a recommendation for early switch therapy. The objective is to demonstrate that the rule is eventually learned from these alerts.

We created two datasets of different sizes and ratios of positive sequences. We created the first dataset with patients who received piperacillin-tazobactam (TAZO), our centre's most prescribed intravenous antibiotic. We created another smaller dataset with patients who received metronidazole (METRO), an ATM predominantly prescribed orally. They were partitioned into *training* and *test* datasets, as described in Table 1.

Table 1. Description of the two datasets used in our experiments

	Dataset	Episodes	Sequence	Positive	Attribute
METRO	Training	9,176	132	12	1206
	Test	19,182	278	46	
TAZO	Training	37,428	485	190	1581
	Test	68,188	947	413	

TIM extracted an accurate and sensitive set of 35 rules. While specificity was lower, it remained above APSS without TIM. A microbiology-infectiology expert evaluated their clinical relevance using a five-point Likert scale ranging from 1-no relevance to 5-excellent relevance. Excellent relevance required the presence of all three indications for early switch therapy. Missing or indirect indications decreased the score. Rules with more than 40 conditions were also penalized. Table 2 presents the scores; 63% of the rules were found to be clinically relevant (score ≥ 3). Interestingly, rules with high relevance scores also had the highest

Table 2. Relevance score of 35 extracted rules (1-no relevance to 5-excellent relevance)

Relevance score	**1**	**2**	**3**	**4**	**5**
#rules	8 (23%)	5 (14%)	8 (23%)	8 (23%)	6 (17%)

information content (*J-measure*). On the other hand, rules with relevance score of 1 were very specific and covered less than 1% of the test set.

Consider the rule in Example 1 with a relevance score of 5. Clear indications for early switch therapy are respected with normalized white cell count (WCC), extended intravenous (IV) treatment, and concurrent oral treatment. This rule also contains complementary information that was used by our expert to extract profiles of patients associated with early switch recommendations. Prolonged stay at the emergency room (ER), old age, *salbutamol*, and additional ATM coverage with *ciprofloxacin* may indicate suspicion of pneumonia caused by resistant pathogens. Ten rules targeted patients under post-operative ATM prophylaxis, a practice not supported by medical evidence that will be addressed by our ATM stewardship team. Another finding was that eight rules targeted patients with $BMI \geq 40$. It could suggest that extended intravenous treatments are prescribed for very severely obese patients to ensure targeted concentrations are achieved. This new information provides insight into our centre's prescribing practices. These patient profiles are of high interest for further investigation as they identify subgroups of patients that could require closer monitoring or wards that could benefit from targeted in-service training.

Example 1. $\{< gender, F, 0, 70 >, < age, 83, 0, 70 >, < ward, ER, 0, 70 >, < WCC, normal, 0, 70 >, < neut, high, 0, 70 >, \{tazocin < dose, 3000, 0, 70 >, < freq, 6, 0, 70 >, < route, IV, 0, 70 >\}, \{ciprofloxacin < dose, 400, 0, 23 >, < freq, 12, 0, 23 >, < route, IV, 0, 23 >\}, \{acetaminophen < dose, 650, 0, 70 >, < freq, 12, 0, 70 >, < route, oral, 0, 70 >\}, \{salbutamol < dose, 0.5, 0, 70 >, < freq, 24, 0, 70 >, < route, inhaled, 0, 70 >\}\} \Longrightarrow true$

We also compared TIM to three other algorithms (see Table 3) to evaluate its relative recall, accuracy and computation times. The first was an instance-based learning (IBL) algorithm that performs nearest neighbor classification with every positive sequence of the training set. The second was a classification rule learner (CRL) using a specialization approach with the *J-measure*, removing positive sequences covered by newly created rules. The third algorithm used an association rule mining (ARM) approach. Various strategies were used in CRL and ARM to focus on highly predictive rules for the positive class. For example, ARM used candidate pruning on both support (METRO: ≥ 0.015; TAZO: ≥ 0.02) and confidence ($conf \geq 0.75$), and eliminated dominated patterns [6]. We restricted ARM to a maximum rule size of 4 for the TAZO test.

Overall, TIM achieved relatively similar or better recall and accuracy than CRL and IBL, except for the recall metrics in METRO, where TIM is outperformed by IBL. TIM achieved superior accuracy than IBL with fewer rules, and

Table 3. Compared results of TIM, IBL, CRL, and ARM on two datasets

Dataset	Method	#rules	Time (s)	Precision	Recall	Accuracy
METRO	TIM	5	0.4	53.8	76.1	85.3
	IBL	12	0.2	30.1	95.7	62.6
	CRL	1	46.0	56.5	76.1	86.3
	ARM	8 074	15.2	44.1	32.6	82.0
TAZO	TIM	30	73.5	62.5	99.0	73.7
	IBL	190	9.5	59.3	99.3	70.0
	CRL	6	583.1	71.4	88.1	79.4
	ARM	614 652	17 864.5	66.0	81.4	73.6

80% and 30% less conditions per rule for METRO and TAZO, respectively. TIM was 7 to 100 times faster than CRL. As can be seen in Table 3, ARM demonstrated the worst results and required heavy post-processing to identify a subset of accurate rules.

6 Conclusion and Future Work

The main motivation of this work is to automatically improve the knowledge base of an antimicrobial prescription monitoring system (APSS) by using supervised machine learning. The system analyzes prescriptions and produces alerts on seemingly inappropriate prescriptions. The rejections of alerts by the pharmacist and the physician provide feedback for a supervised machine learning algorithm (TIM) that learns new rules for the knowledge base.

TIM is still in the experimental stage. It combines instance-based learning and rule induction to learn prescription classification rules from feedback. We have discussed preliminary results showing TIM's capability of learning rules for appropriate early switch from intravenous to oral antimicrobial therapy. The majority of learned rules were found to be clinically relevant because they succeeded in identifying the clinical indications for early switch therapy. A clinician identified from these rules patient profiles associated with early switch recommendations providing further insight into our center's prescribing practices and a potential for targeted interventions (e.g., unsupported use of post-operative antimicrobial prophylaxis). TIM's learning capability aims to extract rules for evaluating treatments that must be adjusted to the patient's evolving clinical condition; we believe it could be extended to many other treatments.

We find these preliminary results very promising. The next steps are pursued experimentation of TIM before its release with the currently deployed version of APSS. The release version will require tools for assisting physicians in revising the rules learnt by TIM before they are incorporated in the knowledge base.

Acknowledgements. This project was partially funded by the *Fonds de recherche du Québec – Santé*, the *Fonds de recherche du Québec – Nature et technologies*, and the *Natural Sciences and Engineering Research Council of Canada*.

References

1. Dellit, T.H., Owens, R.C., McGowan, J.E., Gerding, D.N., Weinstein, R.A., Burke, J.P., Huskins, W.C., Paterson, D.L., Fishman, N.O., Carpenter, C.F., Brennan, P.J., Billeter, M., Hooton, T.M.: Infectious Diseases Society of America and the Society for Healthcare Epidemiology of America guidelines for developing an institutional program to enhance antimicrobial stewardship. Clin. Infect. Dis. 44(2), 159–177 (2007)
2. Valiquette, L., Cossette, B., Garant, M.P., Diab, H., Pepin, J.: Impact of a reduction in the use of high-risk antibiotics on the course of an epidemic of clostridium difficile-associated disease caused by the hypervirulent nap1/027 strain. Clin. Infect. Dis. 45(suppl. 2), 112–121 (2007)
3. Shahar, Y.: A framework for knowledge-based temporal abstraction. Artif. Intell. 90(1-2), 79–133 (1997)
4. Bellazzi, R., Larizza, C., Magni, P., Bellazzi, R.: Temporal data mining for the quality assessment of a hemodialysis service. Artif. Intell. Med. 34(1), 25–39 (2005)
5. Concaro, S., Sacchi, L., Cerra, C., Fratino, P., Bellazzi, R.: Mining healthcare data with temporal association rules: Improvements and assessment for a practical use. In: Combi, C., Shahar, Y., Abu-Hanna, A. (eds.) AIME 2009. LNCS, vol. 5651, pp. 16–25. Springer, Heidelberg (2009)
6. Zaki, M., Lesh, N., Ogihara, M.: Planmine: Predicting plan failures using sequence mining. Artif. Intell. Rev. 14(6), 421–446 (2000)
7. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th Int. Conf. on Very Large Data Bases, VLDB 1994, pp. 487–499. Morgan Kaufmann Publishers Inc., San Francisco (1994)
8. Hartge, F., Wetter, T., Haefeli, W.E.: A similarity measure for case based reasoning modeling with temporal abstraction based on cross-correlation. Comput. Methods Programs Biomed. 81(1), 41–48 (2006)
9. Montani, S., Portinale, L., Leonardi, G.: Case-based retrieval to support the treatment of end stage renal failure patients. Artif. Intell. Med. 37(1), 31–42 (2006)
10. Domingos, P.: Unifying instance-based and rule-based induction. Machine Learning 24(2), 141–168 (1996)
11. Chawla, N., Japkowicz, N., Kotcz, A.: Editorial: Special issue on learning from imbalanced data sets. ACM SIGKDD Explorations Newsletter 6(1), 1–6 (2004)
12. Smyth, P., Goodman, R.M.: Rule induction using information theory. In: Piatetsky-Shapiro, G., Frawley, W.J. (eds.) Knowledge Discovery in Databases, pp. 159–176. AAAI/MIT Press (1991)