A Methodology to Reduce the Complexity of Validation Model Creation from Medical Specification Document

Francesco Gargiulo, Stefano Silvestri, Mariarosaria Fontanella and Mario Ciampi
Institute for High Performance Computing and Networking, ICAR-CNR, Via Pietro Castellino 111 - 80131, Naples, Italy

Keywords: Clustering, Medical Specification Document, Validation, Natural Language Processing (NLP), Schematron.

Abstract: In this paper we propose a novel approach to reduce the complexity of the definition and implementation of a medical document validation model. Usually the conformance requirements for specifications are contained in documents written in natural language format and it is necessary to manually translate them in a software model for validation purposes. It should be very useful to extract and group the conformance rules that have a similar pattern to reduce the manual effort needed to accomplish this task. We will show an innovative cluster approach that automatically evaluates the optimal number of groups using an iterative method based on internal cluster measures evaluation. We will show the application of this method on two case studies: i) Patient Summary (*Profilo Sanitario Sintetico*) and ii) Hospital Discharge Letter (*Lettera di Dimissione Ospedaliera*) for the Italian specification of the conformance rules.

1 INTRODUCTION

The availability of medical information processing systems and the digitalization of almost all information in hospital and clinical processes provide an important support for the tasks of healthcare professionals. Dealing with digital documents and using custom processing systems can improve their work, offering a lot of innovative tools and instruments, ranging from improved information retrieval systems to intelligent image and text processing.

Focusing especially on text documents, we know that an important part of the work of healthcare professionals is the editing of many different clinical documents such as Patient Summaries, Laboratory Tests Reports and Medical Prescriptions. All of them are structured or semi-structured text documents and, furthermore, they even require the presence of certain information, like, for example, a doctor name, a date or a disease code. In addition, their structure and content must respect official guidelines, often established by law. These specifications propose to standardize the structure of these digital documents, ensuring the correctness and the completeness of the content and of the text format.

A standard like the ones promoted by HL7 not only can ensure the semantic and formal correctness of the digital version of these documents, but it supports an effective and reliable automatic processing and the interoperability between different systems too (Ciampi et al., 2016). In other words, it is crucial that the exchanging of these documents between different hospitals or physicians is error-free, without loss of information.

Due to the importance of these tasks, the definition of the conformance rules is a long and critical process, that involves many specialists from medical, clinical, legal and computer science fields and, of course, the governments and health-care agencies. The results of their work are usually documents written in natural language, containing a set of conformance requirements rules for specifications that define the format and the content each of them.

The need of conformance rule documents arises from requirements of standards. Official medical natural language text documents must not only be automatically processed easily, but even respect a format and contain specific information. In Italy, Agencies and government representatives, at this aim, have produced the conformance specifications documents for the digital version of the Patient Summary, the Laboratory Medicine Report, the Hospital Discharge Letter and the Medical Prescription, that are actually part of HL7 International standards in the Italian context.

As explained before, the specifications are documents written in natural language format, describing the whole conformance rules for specifications and the details of the implementation guide for each of

Gargiulo F., Silvestri S., Fontanella M. and Ciampi M.
A Methodology to Reduce the Complexity of Validation Model Creation from Medical Specification Document.
DOI: 10.5220/0006291404970507

digital medical certificates listed above.

Among all the possible uses of conformance rules, one of them could be the development of a validation model, that ensures and tests the complete conformance of the digital certificate to the standard statements.

To implement this kind of functions, computer scientists and engineers must perform a long and tedious task, analysing the natural language text in the conformance specifications document to realize a complete and reliable validation schema for each rule listed in the standard (Gargiulo et al., 2016). This task can be performed only by an hand-made translation of each natural language rule in a software model for validation purposes, using, for example, Schematron (ISO/IEC 19757-3:2016) (Jelliffe, 2001), or other rule-based validation languages. Nowadays, it is a critical task to extract automatically a validation schema from a set of rules described in natural language.

A great boost in the realization of the validation schema can be obtained simply reducing the complexity of the problem, decreasing the number of assertions that has to be manually built. This task can be accomplished grouping the rules following the same pattern: in this way, the same assertion function could be applied to more rules, speeding up the development of the validation model.

In this paper we propose an innovative methodology based on unsupervised machine learning techniques, namely clustering, that extracts automatically the text of the rules from the specification documents and groups them together. Each group contains all the rules that belong to the same assertion schema. The experiments have been performed on Italian language specification rule documents of medical topic, but the proposed techniques are language independent and they can be applied on documents in different languages, or to any kind of specification document.

The paper is structured as follow: in Section 2 it will be given a critical review of the state of the art related to automatically validation and clustering optimization fields; in Section 3 it will be shown the methodology and in Section 4 it will be detailed the designed architecture; in Section 5 the methodology correctness will be demonstrated for two use cases: i) *Patient Summary* and ii) *Hospital Discharge Letter*. Finally, in Section 6 it will be given the conclusion and it will be draw up some key issues for future works.

2 RELATED WORKS

Nowadays there is a big interest of scientists about creation and automatic validation of conformance rules in natual language, especially for medical domain. In (Boscá et al., 2015) the authors proposed and described the archetypes to generate a rules in Natural Language text and Schematron rules for the validation of data instances. The goal was creating a formal document with a formal value of archetype, but at same time understandable by non-technical users.

In (Boufahja et al., 2015) the authors demonstrated the conformance of their samples with HL7 CDA requirements and evaluated the capability of the tools to check those requirements. They looked at the conformance of the provided samples with the basic HL7 CDA requirements as specified within the *Clinical Document Architecture, R2 Normative Edition*, and analysed the capability of the tools provided to check the requirements. At the first time, the authors revisited the CDA specifications and extract the requirements not covered by the CDA Schema, then they checked the coverage of the requirements with another validation tools.

In (Hamilton et al., 2015) the authors described a method in which users realize the benefits of a standards-based method for capturing and evaluating verification and validation (V&V) rules within and across metadata instance documents. The rule-based validation and verification approach presented has the primary benefit that it uses a natural language based syntax for rule set, in order to abstract the computer science-heavy rule languages to a domain-specific syntax. As a result, the domain expert can easily specify, validate and manage the specification and validation of the rules themselves.

In (Jafarpour et al., 2016) is evaluated the technical performance and medical correctness of their execution engines using a range of Clinical Practice Guidelines (CPG). They demonstrated the efficiency of CPG execution engines in terms of CPU time and validity of the generated recommendation in comparison to existing CPG execution engines.

Clustering is an unsupervised machine learning technique, that can well group together objects that show similarity between each others. One of the main problem in clustering, being unsupervised, is the cluster validation, that, in fact, has long been recognized as one of the crucial issues in clustering applications. Validation is a technique to find a set of clusters that best fits natural partitions without any class information, finding the optimal number of clusters (Halkidi and Vazirgiannis, 2001).

The measures used for cluster validation purposes

can be categorized into two classes: external and internal. The first case can be used when a gold case is available, verifying the correctness of results through measures like F-measure, Entropy, Purity, Completeness, Homogeneity, Jaccard coefficient, Fowlkes and Mallows index, Minkowski Score and others (Rendón et al., 2011), (Wu et al., 2009), (Handl et al., 2005), (Rosenberg and Hirschberg, 2007). These papers analysed and compared all the aspects of each measure to understand how well it fits specific cluster algorithm, application or topic, revealing the goodness of the clustering. A common ground of external measures is that they can often be computed by the contingency matrix (Wu et al., 2009).

When a gold case is not available, the second class of cluster validation measures, namely the internal ones, must be used. In this case, the goodness of clustering results is based only on spatial characteristics of cluster members, like their compactness or separation. One of the first internal cluster measure proposed in literature is the silhouette (Rousseeuw, 1987). The silhouette is a numeric parameter that takes in account the tightness and separation of each cluster, showing which objects lie well within their cluster and which ones are merely somewhere in between clusters. Many other internal measures have been defined in literature, like Dunns indices, SD and SD_bw validity indexes and others (Liu et al., 2010), taking into account different aspects of the clustering results in addition to the separation and compactness, like monotonicity, noise, density, sub-clusters and skewed distributions, that can better show different aspects of the results.

Internal cluster measures have been often used to set the correct cluster number, not only optimizing their global value (Kaufman and Rousseeuw, 2009), but even obtaining some specific new measures from the classical ones, to identify cluster characteristics of a specific domain, as, for example, they did in (Pollard and Van Der Laan, 2002). In (Dhillon et al., 2002) an iterative clustering method is proposed to improve spherical *K*-means algorithm results, that, when applied to small cluster sizes, can tend to get stuck at a local maximum far away from the optimal solution. They presented an iterative local search procedure, which refines a given clustering by incrementally moving data points between clusters, thus achieving a higher objective function value.

3 METHODOLOGY

In this Section we explain the details of the methodology applied in our experiments. We developed an iterative cluster strategy, that aims to obtain the best clustering solution. This is achieved through an internal measure cluster selection, described in 3.1 and in 3.2. Then, to assess the whole methodology, we manually built a gold case, validated using a custom cluster external validation measure described in 3.3. Gold case construction and validation assessment are described in Section 5

3.1 Clustering Algorithm and Internal Measures

Following the literature (Alicante et al., 2016a), we decided to use the spherical *K*-means cluster algorithm, a slight variation of the *K*-means algorithm, and the *cosine* similarity. It has been shown that the optimal distance for *K*-means based cluster applications for Italian natural language text of medical topic is the *cosine distance* (Alicante et al., 2016b), that is equals to inverse cosine similarity (eq. 1).

$$1 - \sum_{i=1}^{M} \frac{x_i \cdot y_i}{\mid x_i \mid\mid y_i \mid} \tag{1}$$

The cosine similarity measure allows to use the spherical *K*-means (Zhong, 2005) algorithm, that uses a slight variation of the *K*-means algorithm exploiting the cosine similarity measure: the classical *K*-means minimizes the mean squared error from the cluster centroid (eq. 2)

$$\frac{1}{N} \sum_{\mathbf{x}} \|\mathbf{x} - \mu_{k(\mathbf{x})}\| \tag{2}$$

where N is the total number of feature vectors and $\mu_{k(\mathbf{x})}$ is the most similar centroid; instead, in spherical *K*-means the objective function is defined as (eq. 3)

$$\sum_{\mathbf{x}} \mathbf{x} \cdot \mu_{k(\mathbf{x})} \tag{3}$$

that is strictly related to the cosine similarity. Our experiments confirm the goodness of these choices (see Section 5).

The determination of optimal partition is performed through an iterative loop, based on cluster internal measure, described in details in Section 3.2.

As assessment of clustering results we can only use internal measures, having no labelled data. For validation purposes we have chosen the *silhouette* (Rousseeuw, 1987), a classic cluster internal validation measure, that takes into account two important aspects of a clustering solution: i) the similarity among elements of the same cluster and ii) the dissimilarity among elements belong to different clusters.

Let call i a generic point of the data set and $a(i)$ the average dissimilarity of the point with the elements

of the same cluster. Dissimilarity is calculated with inverse cosine similarity. A small $a(i)$ means that the point is quite close to all the other points in the cluster. We define $b(i)$ as the smallest average dissimilarity between i and the elements of any cluster different from the one i belongs, estimating how far the current point is from the closest point not in the same cluster.

Then the silhouette $s(i)$ of each point of a cluster is defined as:

$$s(i) = \frac{b(i) - a(i)}{max(a(i), b(i))} \quad (4)$$

where the opposite of $a(i)$ is considered so that its effect is in favour of compactness. The silhouette value is in the range $[-1, 1]$, and a larger silhouette indicates a better assignment of that point to its cluster. The silhouette is negative whenever the other points in the cluster are, on average, farther from the point i than the closest point outside of the cluster. Silhouette can then be averaged on all points of a cluster to assess the compactness of that cluster with respect to the others. In this case, a negative number of silhouette means that the diameter of the cluster is larger than the distance from the closest point out of the cluster.

The average silhouette over all elements i could be used to compare clustering results and to select the optimal number of clusters k by maximizing it over a range of possible values for k (Kaufman and Rousseeuw, 2009). The method of maximizing average silhouette can be used with any clustering algorithm and any distance metric, but it has the disadvantage that measures only the global structure of the solution. To take in account finer behaviour we have proposed an alternative parameter. Let consider the average silhouette of the j^{th} cluster as S_j. We then call *MAS* the median of average silhouettes $S = \{S_1, S_2, \ldots, S_k\}$, a value equals to:

$$MAS = median(S) \quad (5)$$

The *MAS* can give a synthetic clue about the goodness of the entire cluster solution, but, differently from the simple average of all silhouettes $s(i)$, it can take into account each cluster validity.

3.2 Iterative Cluster Optimization

To obtain a more precise clustering we proposed an iterative clustering optimization algorithm, based on *MAS* optimization. In Figure 1 is depicted flow chart diagram, representing the proposed methodology.

After constructing a Vector Space Model (VSM) of the Conformance Rules, we have defined an iterative cycle. The first task is a de-noising of the input data, using a Principal Component Analysis

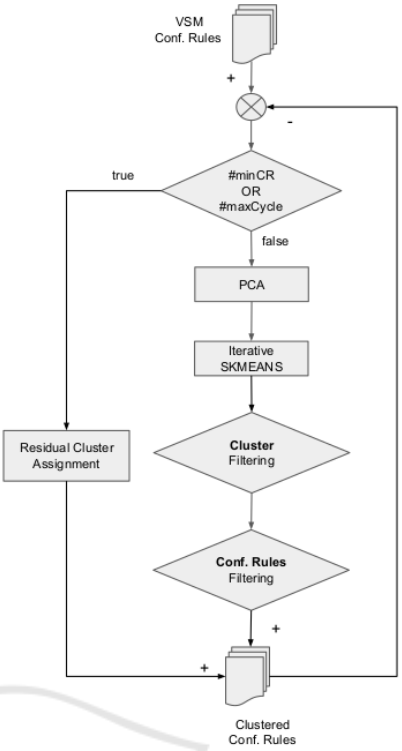

Figure 1: Flow chart of the methodology used to cluster the conformance rules.

(PCA) methodology for feature reduction (Cao et al., 2003). We set the selection of information content of PCA at 96%: this value has been obtained observing the higher mean silhouette value of clustering experiments. The feature reduction is performed at each step of the iterative cluster algorithm, reducing each iteration the number of extracted features.

Then, an Iterative Spherical K-means algorithm, depicted in Figure 2, is applied to evaluate the optimal cluster solution. We perform Spherical K-means with cluster number ranging from 2 to total Conformance Rules number. The optimal cluster solution is the one with highest *MAS* value in the range of all solutions obtained during the iteration. From the whole solution with highest *MAS* we select only clusters whose mean silhouette is bigger than *MAS* (*inter cluster selection*); then, in these selected clusters, we filter out the elements whose silhouette is smaller *MAS* (*intra cluster selection*). The clusters obtained with this filtering operations are selected as part of the final solution and the remaining elements are iteratively re-processed in the same way, until the number of remaining documents is smaller than a given threshold *#minCR* or the number of iterations is bigger than a threshold *#maxCycle* (see Figure 1).

When the termination condition is reached, the

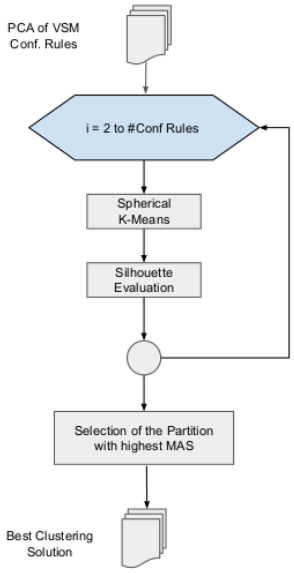

Figure 2: Flow chart of the implemented Iterative Spherical K-Means.

residual conformance rules are assigned to their own cluster (one element cluster), considering that each rules have a low silhouette value.

3.3 Cluster Validation Measure

The evaluation of the clustering goodness considering an handmade gold case is obtained using external measures, that are often computed by the contingency matrix (Wu et al., 2009).

The contingency matrix (see Tab.1) is defined as follow: given a data set D with n objects, assume that we have a partition $C = \{C_1, \ldots, C_{K'}\}$ of D, where $\bigcup_{i=1}^{K'} C_i = D$ and $C_i \bigcap C_j = \emptyset$ for $1 \leq i \neq j \leq K'$, and K' is the number of clusters. If we have a *Gold Case*, we can have another partition on $D : P = \{P_1, \ldots, P_K\}$, where $\bigcup_{i=1}^{K} P_i = D$, $P_i \bigcap P_j = \emptyset$ and K is the number of classes. Each element n_{ij} of the matrix denotes the number of objects in cluster C_i from class P_j.

Table 1: The Contingency Matrix.

	C_1	C_2	...	$C_{K'}$	$\sum$
P_1	n_{11}	n_{12}	...	$n_{1K'}$	$n_{1\cdot}$
P_2	n_{21}	n_{22}	...	$n_{2K'}$	$n_{2\cdot}$
.	.	.	...	.	.
P_K	n_{K1}	n_{K2}	...	$n_{KK'}$	$n_{K\cdot}$
$\sum$	$n_{\cdot 1}$	$n_{\cdot 2}$	...	$n_{\cdot K'}$	n

From the contingency matrix it is possible to define for each obtained cluster C_j and for each gold case cluster P_i the following two measures (Rosenberg and Hirschberg, 2007):

- **Homogeneity** $Hom(C_j)$**:** a clustering must assign only those data-points that are members of a single class to a single cluster. It can be calculated as:

$$Hom(C_j) = \frac{1}{n_{\cdot j}} \max_j(n_{ij}) \tag{6}$$

- **Completeness** $Com(P_i)$**:** a clustering must assign all of those data-points that are members of a single class to a single cluster. Completeness is symmetrical to Homogeneity.

$$Com(P_i) = \frac{1}{n_{i\cdot}} \max_i(n_{ij}) \tag{7}$$

These two measures are both needed to characterize the goodness of the clustering partition, taking into account two complementary aspects. Using them, we defined a new measure for the whole dataset partition named as *Clustering Goodness* (*CG*) defined as the weighted mean of the *Hom* and *Com* (see eq. 8). The weighting is necessary because the goodness of cluster solution is related even to the correct choice of cluster number and not only to *Hom* and *Com*. In other words, if clusters number is close to documents number, the $mean(Hom)$ value tends towards one; on the other hand, if the clustering solution is made by only one cluster, the $mean(Com)$ tends towards one. This extreme cases demonstrate that an arithmetic mean of these measures does not capture clustering goodness in every case.

$$CG(C) = \frac{1}{K+K'} \sum_{i=1}^{K} \sum_{j=1}^{K'} \alpha \cdot Com(P_i) + (1-\alpha) \cdot Hom(C_j) \tag{8}$$

The α value must balance the negative effects previously described, taking into account the cluster number in function of the gold-case cluster number. So we defined α as:

$$\alpha = \begin{cases} \frac{1}{2 \cdot K'}, & \text{if } K \leq K' \\ \frac{1}{2 \cdot (n-K')}, & \text{otherwise} \end{cases} \tag{9}$$

In this way, the value of equation 8 varies in range $(0, 1]$ and a perfect clustering solution has a *CG* value equal to 1, meaning that the clustering is identical to gold case partition, but the α value as defined in 9 can weight the importance of *Hom* and *Com* in function of optimal cluster number too. We used the *CG* in the experimental assessment in Section 5, showing the effectiveness of the proposed methodology.

4 SYSTEM ARCHITECTURE

The system architecture is divided into six different blocks, as shown in Figure 3.

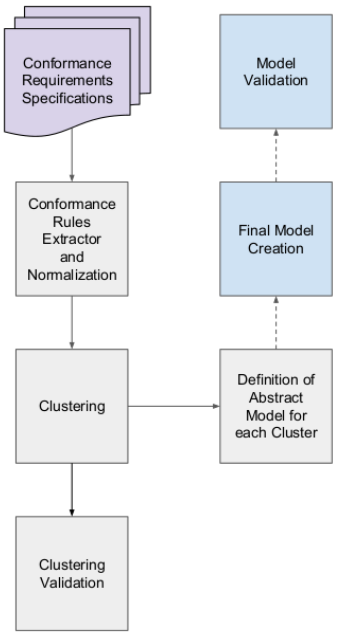

Figure 3: Main System Architecture. In purple, the input data; in grey, the blocks that are evaluated in this paper and in light-blue the blocks that will be considered as future works.

The first block consists in a pre-processing stage where the input specification document is converted into a more structured file, extracting and normalizing the conformance rules from the text. In the second block, a vector space model is created, extracting the features from the text of conformance rules and the iterative clustering technique, previously described, is applied, to obtain the group of rules that respect the same pattern. In the third block, a clustering evaluation is made considering hand-made gold cases. In the fourth block, an implementation of an abstract rule for each cluster is defined. In the fifth block we plan to create a module that implements each conformance rule according with its own abstract rule and, finally, in the last block is planned to evaluate the correctness of the *final* model obtained using hand-made gold cases.

The whole pipeline has been implemented in *Knime* environment (Berthold et al., 2007), an open platform for machine learning that natively supports external scripts in the most common language (C, Python, Java, R, Matlab, Weka and other). Using Knime is possible to integrate many tools in a single environment and design an optimized pipeline. In Figure 4 is shown the workflow implemented for the experiments.

The following subsections 4.1, 4.3, 4.4 describe the details of each block and the tools used to realize the system.

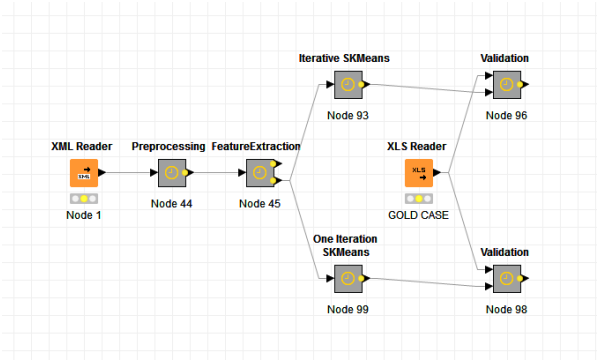

Figure 4: Example of Knime workflow. The grey blocks represent a *metanode* that is a group of nodes which perform complex operations.

4.1 Conformance Rule Extraction and Normalization

The first task of the system is the extraction of rules text and its normalization, obtained after an analysis of documents and conformance text structure. A specification document is often in pdf file format and the first operation needed to perform any kind of processing is the conversion in a plain-text *UTF-8* file.

We converted the conformance rule documents used in our experiment using *pdftotxt*[1], an open source command-line utility, included by default within many Linux distributions. The text file outputted from *pdftotxt* preserves most of the indentation and the structure of the original pdf file and this is really important for the subsequent task of the system. In fact, we need to extract only the natural language text where each rule is defined and stated. This task has been accomplished writing some Python scripts that, using the regular patterns of the text, extract the index of the document, the paragraph names and the rule texts. In our case, for example, all the conformance rules have a well-defined pattern: their definitions start with a tag (CONF-xx-yy) and end with another rule tag (CONF-xx-zz) or a dot sign. The next Figure 5 shows an example of the regular pattern from the original conformance rules file.

Different conformance documents can be processed only analysing their specific patterns. Python scripts have in input the start and end rule delimiter, but they can be easily modified to be applied to different and more complex patterns. The scripts perform a normalization phase too, deleting punctuation, the not-standard characters, symbols and stop words (a list of Italian language stop words is provided by Lucene[2]). Then, using regular expressions,

[1] http://www.foolabs.com/xpdf/home.html

[2] https://lucene.apache.org/core/4_4_0/analyzers-common/org/apache/lucene/analysis/it/ItalianAnalyzer.html

DEVE essere presente un elemento *patient/name* contenente nome e cognome del paziente (vedi § 2.3 - Persone ed Organizzazioni). Non può essere utilizzato il nullFlavor per indicare l'indisponibilità del dato.

CONF-14: L'elemento *patientRole/patient* **DEVE** contenere l'elemento *patient/administrativeGenderCode* (sesso).

CONF-15: L'elemento *patientRole/patient* **PUÒ** contenere l'elemento *patient/birthTime* (data di nascita).

CONF-16: L'elemento *patientRole/patient* **PUÒ** contenere l'elemento *patient/birthPlace/place/addr/censusTract* che riporta il codice ISTAT del luogo di nascita dell'assistito.

Per i dettagli relativi agli elementi *patient/name patient/administrativeGenderCode* e *patient/birthTime* si rimanda a Rif 10.

Figure 5: An example of input document used for experiment assessment. It is possible to observe the regular pattern of conformance rules to be extracted. In addition, each rules lies on a grey background.

we replace *Logical Observation Identifiers Names and Codes* (LOINC[3]), TemplateId codes, Paragraphs and Key Names with a generic identifier (i.e. LOINC $33882-2$ is substituted with the word *LOINC*). We did this further normalization to reduce the noise induced by different terminology associated to the same concept, obtaining a better clustering results.

The output of this module is an xml file, whose structure is depicted in Figure 6. As shown, the body of CONF tags contains only the normalized text of each conformance rule. The used tags are the following:

- *documento*: the xml root, its body contains the document title and all the paragraphs will be its children;
- *paragrafo*: it contains the paragraph name in its body and the paragraph number as *id* attribute. All the CONF associated to it are its own children;
- *CONF*: it contains in its body the normalized text of the original conformance rule. Its attributes are: i) *num* in which is indicated the rule number and ii) *par* that indicates the paragraph number.

```
<paragrafo id="2.4.2.1.3">
    2.4.2.1.3 Paziente (Human Patient)
    <CONF num="12" par="2.4.2.1.3">-12: L'elemento paragrafo DEVE contenere almeno
    un elemento id con §chiave§ valorizzato a §template§
    ed in cui nell'attributo §chiave§ è riportato il Codice Fiscale del
    soggetto, oppure con §chiave§ valorizzato a "[OID ROOT STP REGIONALI]" ed in
    cui nell'attributo §chiave§ è riportato il Codice STP, oppure con
    §chiave§ valorizzato a §template§ ed in cui
    nell'attributo §chiave§ è riportato il Numero di Identificazione Personale TEAM.
    </CONF>
    <CONF num="13" par="2.4.2.1.3">-13: Il documento DEVE contenere l'elemento
    paragrafo DEVE essere presente un elemento §context§
    contenente nome e cognome del paziente (vedi paragrafo 2.
    3 Persone ed Organizzazioni). Non può essere utilizzato il nullFlavor per
    indicare l'indisponibilità del dato.</CONF>
    <CONF num="14" par="2.4.2.1.3">-14: L'elemento
    paragrafo DEVE contenere l'elemento
    §context§ (sesso).</CONF>
    <CONF num="15" par="2.4.2.1.3">-15: L'elemento paragrafo PUÒ
    contenere l'elemento paragrafo (data di nascita).</CONF>
    <CONF num="16" par="2.4.2.1.3">-16: L'elemento
    paragrafo PUÒ contenere l'elemento
    §context§ che riporta il codice ISTAT del luogo
    di nascita dell'assistito.</CONF>
</paragrafo>
```

Figure 6: Part of the output xml file obtained from text extraction and normalization module.

[3]http://loinc.org/

4.2 Feature Selection

The input to machine learning applications is represented through a *Vector Space Model* (VSM). In VSM a vector is associated to each sample (in this case the Conformance Rule) in which the elements of the vector correspond to the feature values.

Vectors of size M correspond to points in an M-dimensional space; the main hypothesis underlying the VSM is that similar objects are represented by points which are closed in the M-dimensional space. Achieving optimal results with a machine learning technique based on VSM is strictly related to the correct choice of feature space (Amato et al., 2013).

In our case, the entity to be clustered are the conformance rules in natural language text, identified by their name. The rules in a VSM are mapped as n-grams of words. The correct selection of the n-gram size, namely the length of n, is both language and topic dependant (Cavnar and Trenkle, 1994) and so there is not an absolute rule (Eder, 2011). In our case we selected all n-grams with n ranging from 2 to 6, observing the highest *MAS* (see equation 5) obtained in different clustering experiments, varying both n and the number of n-grams together. The high value of n obtained (often only uni-grams, bi-grams and tri-grams are used in literature) can be explained by the repetitive structure of the patterns in the description of a rule. We extract the features using internal KNIME modules.

The VSM obtained can be represented by a high dimensional sparse matrix. To reduce the noise caused by not discriminant features and consequently the space dimension, improving clustering performance and providing a faster computation, we applied Principal Component Analysis (PCA) as feature reduction method. The PCA has been implemented through the Cran R built in function *prcomp*, a really fast and accurate PCA algorithm. We set the selection of information content of PCA at 96%: this value has been obtained observing the higher mean silhouette value within all clustering experiments. The feature reduction is performed at each step of the iterative cluster module, described in the next Section 4.3, reducing each time the number of extracted features.

The use of n-grams directly extracted from the dataset makes the whole process totally language independent; the same methodology can be applied on conformance rules in any language and even to mixed languages, or medical slang documents. Changing the input dataset affects only the scripts for the normalization and rule extraction, that must be slightly modified as described in previous Section 4.1, but none of the other modules, included the feature extraction one.

4.3 Iterative Clustering

As described in Section 3, to group the conformance rules we applied iteratively a spherical *K*-means algorithm, selecting at each step the best solution according to *MAS* (equation 5), a cluster internal measure based on silhouette. After applying PCA feature reduction, we used the Cran R *skmeans* package (Hornik et al., 2012) with CLUTO algorithm (Karypis, 2002) to iteratively calculate spherical *K*-means with a cluster number range between 2 and the total number of rules, as described in Section 3.2. To speed up the iterative clustering process we used the *doParallel* Cran R package (Weston and Analytics, 2014), running more cluster processes in parallel.

4.4 Abstract Model Definition

The last implemented module performs the abstract model definition. At this aim we use a functionality of the standard Schematron that allows to define abstract patterns. In this way, it is possible to implement for each obtained cluster only one abstract model, obtaining a reduction of the complexity evaluable as:

$$\Delta(\text{Complexity}) = \left(1 - \frac{\text{Cluster Number}}{\text{Conf. Rules Number}}\right) \cdot 100 \quad (10)$$

The Figure 7 represents the conceptual schema for the creation of a *Final Implemented Rule* starting from a *Clustered Conformance Rule* and an *Abstract Pattern Template*.

An abstract pattern template is a way to generalize a class of possible instances of conformance rules and, like the concept of *Abstract Class* in the Object Oriented paradigm, it is possible to instantiate a specific *Final Implemented Rule* starting from it. The

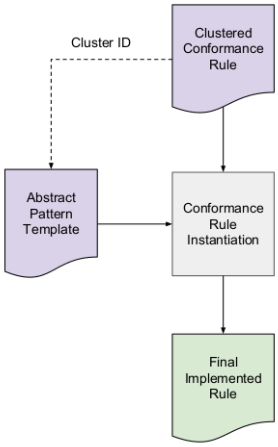

Figure 7: Main Schema of the Conformance Rule Implementation starting from an Abstract Pattern Template and a Clustered Conformance Rule.

```
<sch:pattern abstract="true" id="Cluster_ID">
    <sch:rule id="$Context">
        <sch:assert test= "assertion($par1, …., $parN)> <!— es: count($par1) = 1
            and count($parN) lt&; 2 —>
        Error/Warning: $body"/>
        </sch:assert>
    </sch:rule>
</sch:pattern>
```

Figure 8: Main Schema of the Conformance Rule Implementation starting from an Abstract Cluster Template.

Figure 8 shows a generic example written according to the standard Schematron where, considering a cluster partition identified by *Cluster_ID*, it is created an abstract pattern with all the parameters defined as generic variables (ex. $Context, $par1, $par2, etc.). In the example we also defined a generic function *assertion*$(\cdot)$ to obtain complex tests using the defined variables.

In Figure 9, the abstract pattern is used to instantiate a specific conformance rule that belongs to that cluster. In this case the instantiation consists to declare the abstract pattern to use and to specify each parameter involved.

```
<sch:pattern is-a="Cluster_ID" id="CONF_XXX">
    <sch:param name="$Context" value="value of the context""/> <!—es:cda:patient—>
    <sch:param name= "$par1" value= "value of par1"/> <!— es: cda:code —>
    <sch:param name= "$parN" value= "value of parN"/> <!— es: cda:codeName —>
    <sch:param name="$body" value="body content "/>
</sch:pattern>
```

Figure 9: Main Schema of the Conformance Rule Implementation starting from an Abstract Cluster Template.

At the moment the abstract pattern template implementation is manual and involves a human processing. As future work, we are planning to automatize this task, using NLP tools. In details, the use of a Part of Speech (PoS) tagger and of a dependency parser will automatically identify the subject, the main verb and its objects of each cluster member. In addition, a dedicated entity extraction can help to classify the object types. Then, a rule based system can build a pattern for each cluster.

5 EXPERIMENTAL RESULTS

To verify the effectiveness of described approach we will show the application of the proposed methodology on two case studies, namely the Italian localization of specification of the conformance rules of: i) Patient Summary[4](in Italian *Profilo Sanitario Sintetico*, PSS) and ii) Hospital Discharge Letter[5](in Italian *Lettera di Dimissione Ospedaliera*, LDO). The conformance requirements and specifications are part

[4]Patient Summary: http://www.hl7italia.it/sites/default/files/Hl7/docs/public/HL7Italia-IG_CDA2_PSS-v1.2-S.pdf

[5]Hospital Discharge Letter: http://www.hl7italia.it/webfm_send/1709

Table 2: Gold case cluster number of each dataset.

Specification	Rules number	Gold case Cluster number
LDO	104	42
PSS	259	129
PSS+LDO	363	159

of HL7-Italia (see Section 1 for more details). PSS and LDO are both conformance requirements and specification documents written in semi-structured natural language text. PSS contains 259 conformance rules, while LDO a total number of 104. To extend the experimental assessment, we have applied our methodology even to the sum of the rules from both documents, clustering a new data set with a total of 363 rules, named PSS+LDO. It could be useful in real application group together similar conformance rules documents, identifying the rules with the same patterns from different documents.

The assessment is based on gold cases, formed by the ideal grouping of the conformance rules of each dataset belonging to the same pattern. The goodness of the cluster results have been measured through the *CG* (equation 8) applied on those gold cases. Each gold case has been manually built by the software developers who previously implemented the whole conformance rule validation schema: they well know the rules text and their patterns and so they produced a reliable gold cases for each dataset used. The number of conformance rules grouped in each gold case is shown in next Table 2.

We have compared the results obtained with our approach, namely Iterative Spherical K-Means (IT-SKM), with the ones obtained using a One Iteration Spherical K-Means (1-SKM) method. In this case, only one step of iteration process is performed, choosing the cluster number of the partition with the MAS function (equation 5), without selecting the elements to be clustered in the following steps.

In Table 3 is shown the effectiveness of using IT-SKM for evaluating the optimal number of clusters through the synthetic external measure *CG* (eq. 8), previously defined in Section 3.3. We compared the IT-SKM results with the 1-SKM results through the *CG* measure for PSS, LDO and PSS+LDO cases. In all experiments the best results have been obtained with iterative approach IT-SKM. It is even worth noting that cluster number obtained with IT-SKM is really close to the gold case.

To better understand and explain the results of our experiments, we show in Figure 10 the Hom (in red) and *Com* (in blue) percentage value distribution for 1-SKM and IT-SKM for all data sets. In details, the figures depict the cluster distribution whose Hom and *Com* have a certain value. All 1-SKM experiments have an high number of clusters whose Hom is high, due to the fact that the number of clusters obtained is close to the total conformance rules number and many clusters have only one element. So the high value of Hom is caused simply by cluster formed by only one element, not by a good cluster solution. On the other side, the number of clusters with an high *Com* value is only a little fraction of the whole partition, suggesting a bad clustering.

Instead, IT-SKM experiments show in all cases a very high fraction of clusters with both Hom and *Com* equal to 100%. A perfect solution (identical to gold case) has Hom and *Com* equal to 100% for each cluster. The results in Figure 10 for IT-SKM show that this condition is verified for an high number of clusters, demonstrating the effectiveness of the proposed methodology. In addition, the figure confirms that *CG* measure follows the correct behaviour and it is an useful external measure.

6 CONCLUSION AND FUTURE WORK

In this paper we proposed a novel approach to reduce the complexity of the definition and implementation of a medical document validation model.

We defined an architecture to automatically produce a software specification starting from a set of conformance rules in semi-structured natural language format. At this aim, we presented an innovative cluster approach that automatically evaluates the optimal number of groups using an iterative method based on internal cluster measures evaluation.

The effectiveness of the proposed approach is evaluated on two case studies: i) Patient Summary (*Profilo Sanitario Sintetico*) and ii) Hospital Discharge Letter (*Lettera di Dimissione Ospedaliera*) for the Italian localization specification of the conformance rules.

As future works we are planning to realize the remaining blocks of the architecture depicted in the Figure 3 and, in particular, the *Final Model Creation* and *Model Validation* (the blocks have light-blue background in the Figure). Furthermore, we are considering to automatize the creation of the abstract pattern template starting from a cluster, with the support of natural language tools. At least, we are also investigating more deeply on other unsupervised methods to automatically grouping the conformance rule and in particular on deep-learning approaches.

Table 3: Results. The best results are highlighted in bold.

Specification	Method	Mean(CG)	Mean(COM)	Mean(HOM)	#Cluster	#Gold	#Conf	Δ(Complexity)
LDO	**Iterative**	**74.21%**	70.00%	76.77%	46	42	104	55.77%
	One Iteration	70.85%	53.85%	98.72%	77			25.96%
PSS	**Iterative**	**67.17%**	75.88%	63.53%	108	129	259	58.30%
	One Iteration	65.36%	58.37%	95.48%	211			18.53%
PSS+LDO	**Iterative**	**66.88%**	64.90%	68.16%	167	159	363	53.99%
	One Iteration	60.34%	50.00%	98.43%	313			13.77%

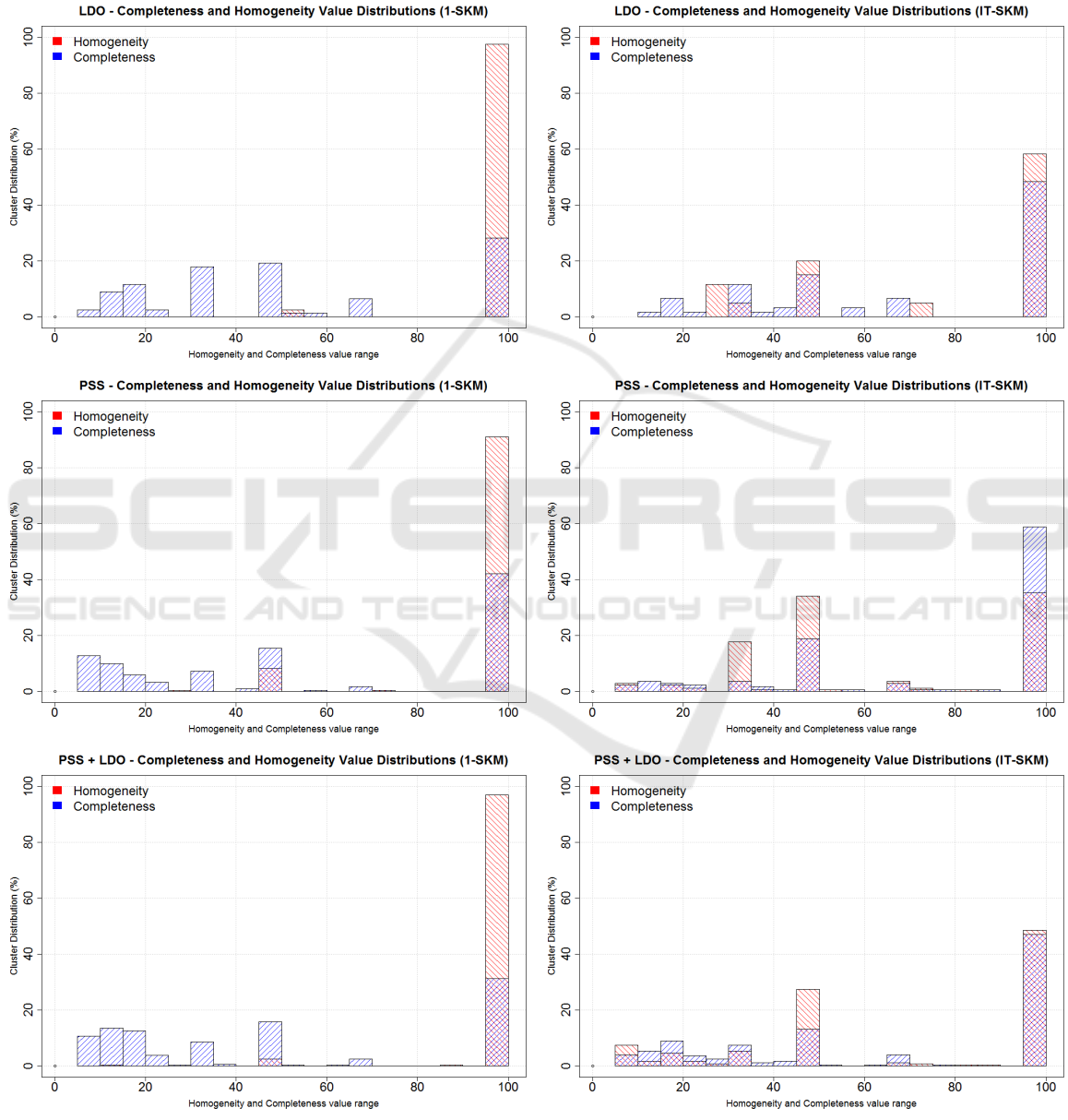

Figure 10: *Hom* and *Com* value distributions for all experimental assessment.

ACKNOWLEDGEMENTS

This work has been partially supported by the Italian project *Realization of services of the national infrastructure for interoperability for Electronic Health Records*, a Convention between the Agency for Digital Italy and the Italian National Research Council.

REFERENCES

Alicante, A., Corazza, A., Isgrò, F., and Silvestri, S. (2016a). Semantic cluster labeling for medical relations. *Innovation in Medicine and Healthcare 2016*, 60:183–193.

Alicante, A., Corazza, A., Isgrò, F., and Silvestri, S. (2016b). Unsupervised entity and relation extraction from clinical records in Italian. *Computers in Biology and Medicine*, 72:263–275.

Amato, F., Gargiulo, F., Mazzeo, A., Romano, S., and Sansone, C. (2013). Combining syntactic and semantic vector space models in the health domain by using a clustering ensemble. In *HEALTHINF 2013 - Proceedings of the International Conference on Health Informatics*, pages 382–385.

Berthold, M. R., Cebron, N., Dill, F., Gabriel, T. R., Kötter, T., Meinl, T., Ohl, P., Sieb, C., Thiel, K., and Wiswedel, B. (2007). KNIME: The Konstanz Information Miner. In *Studies in Classification, Data Analysis, and Knowledge Organization (GfKL 2007)*. Springer.

Boscá, D., Maldonado, J. A., Moner, D., and Robles, M. (2015). Automatic generation of computable implementation guides from clinical information models. *Journal of Biomedical Informatics*, 55:143–152.

Boufahja, A., Poiseau, E., Thomazon, G., and Bergé, A.-G. (2015). Model-based analysis of hl7 cda r2 conformance and requirements coverage. *EJBI*, 11(2).

Cao, L., Chua, K. S., Chong, W., Lee, H., and Gu, Q. (2003). A comparison of pca, kpca and ica for dimensionality reduction in support vector machine. *Neurocomputing*, 55(1):321–336.

Cavnar, W. B. and Trenkle, J. M. (1994). N-gram-based text categorization. *Ann Arbor MI*, 48113(2):161–175.

Ciampi, M., Esposito, A., Guarasci, R., and Pietro, G. D. (2016). Towards interoperability of ehr systems: The case of italy. In *Proceedings of the International Conference on Information and Communication Technologies for Ageing Well and e-Health - Volume 1: ICT4AWE,*, pages 133–138.

Dhillon, I. S., Guan, Y., and Kogan, J. (2002). Iterative clustering of high dimensional text data augmented by local search. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pages 131–138. IEEE.

Eder, M. (2011). Style-markers in authorship attribution a cross-language study of the authorial fingerprint. *Studies in Polish Linguistics*, 6(1):99–114.

Gargiulo, F., Fontanella, M., and Ciampi, M. (2016). Validazione di documenti sanitari strutturati in hl7 cda rel. 2.0 con schemi schematron. Technical report, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR) del Consiglio Nazionale delle Ricerche (CNR).

Halkidi, M. and Vazirgiannis, M. (2001). Clustering validity assessment: Finding the optimal partitioning of a data set. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 187–194. IEEE.

Hamilton, J., Darr, T., Fernandes, R., Jones, D., and Morgan, J. (2015). Rule-based constraints for metadata validation and verification in a multi-vendor environment. In *International Telemetering Conference Proceedings*. International Foundation for Telemetering.

Handl, J., Knowles, J., and Kell, D. B. (2005). Computational cluster validation in post-genomic data analysis. *Bioinformatics*, 21(15):3201–3212.

Hornik, K., Feinerer, I., Kober, M., and Buchta, C. (2012). Spherical *k*-means clustering. *Journal of Statistical Software*, 50(10):1–22.

Jafarpour, B., Abidi, S. R., and Abidi, S. S. R. (2016). Exploiting semantic web technologies to develop owl-based clinical practice guideline execution engines. *IEEE Journal of Biomedical and Health Informatics*, 20(1):388–398.

Jelliffe, R. (2001). The schematron assertion language 1.5. *Academia Sinica Computing Center*.

Karypis, G. (2002). Cluto-a clustering toolkit. Technical report, DTIC Document.

Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons.

Liu, Y., Li, Z., Xiong, H., Gao, X., and Wu, J. (2010). Understanding of internal clustering validation measures. In *2010 IEEE International Conference on Data Mining*, pages 911–916. IEEE.

Pollard, K. S. and Van Der Laan, M. J. (2002). A method to identify significant clusters in gene expression data. In *Proceedings of SCI World Multiconference on Systemics, Cybernetics and Informatics*, pages 318–325.

Rendón, E., Abundez, I., Arizmendi, A., and Quiroz, E. (2011). Internal versus external cluster validation indexes. *International Journal of computers and communications*, 5(1):27–34.

Rosenberg, A. and Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, volume 7, pages 410–420.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Weston, S. and Analytics, R. (2014). *doParallel: Foreach parallel adaptor for the parallel package*. R package version 1.0.8.

Wu, J., Xiong, H., and Chen, J. (2009). Adapting the right measures for k-means clustering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 877–886. ACM.

Zhong, S. (2005). Efficient online spherical K-means clustering. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, volume 5, pages 3180–3185.