A Case-Based Reasoning Model Powered by Deep Learning for Radiology Report Recommendation

Elvira Amador-Domínguez[1], Emilio Serrano[1], Daniel Manrique[2], Javier Bajo[1] *

[1] Ontology Engineering Group, Departamento de Inteligencia Artificial, ETSI Informáticos, Universidad Politécnica de Madrid, 28660 Madrid (Spain)
[2] Laboratorio de Inteligencia Artificial, Departamento de Inteligencia Artificial, ETSI Informáticos, Universidad Politécnica de Madrid, 28660 Madrid (Spain)

Received 8 February 2021 | Accepted 29 May 2021 | Published 11 August 2021

Abstract

Case-Based Reasoning models are one of the most used reasoning paradigms in expert-knowledge-driven areas. One of the most prominent fields of use of these systems is the medical sector, where explainable models are required. However, these models are considerably reliant on user input and the introduction of relevant curated data. Deep learning approaches offer an analogous solution, where user input is not required. This paper proposes a hybrid Case-Based Reasoning, Deep Learning framework for medical-related applications, focusing on the generation of medical reports. The proposal combines the explainability and user-focused approach of case-based reasoning models with the deep learning techniques performance. Moreover, the framework is fully modular to fit a wide variety of tasks and data, such as real-time sensor captured data, images, or text, to name a few. An implementation of the proposed framework focusing on radiology report generation assistance is provided. This implementation is used to evaluate the proposal, showing that it can provide meaningful and accurate corrections, even when the amount of information available is minimal. Additional tests on the optimization degree of the case base are also performed, evidencing how the proposed framework can optimize this base to achieve optimal performance.

Keywords

Case-Based Reasoning, Deep Learning, Natural Language Processing, Entity Recognition, Medical Radiology.

DOI: 10.9781/ijimai.2021.08.011

I. Introduction

Deep Learning is currently a fundamental approach in Artificial Intelligence applied to the medical domain. Their applications include image segmentation [1]–[3], 3D image reconstruction [4], [5], and disease diagnosis [6], [7]. While these approaches offer outstanding results, they suffer from a considerable flaw: lack of explainability. This issue is particularly concerning in the medical domain, where it is crucial to understand the inference procedure carried by a model to perform a task. Moreover, deep learning-based approaches require a considerable amount of labelled data to be truly accurate, which may not always be available.

Opposite to this approach, the Case-Based Reasoning (CBR) methodology [8], [9] provides computational models closely related to human reasoning. In CBR, the resolution of problems provides knowledge that permits to solve new, similar ones. A CBR model discovers the closest situation to the current one to solve and adapt its solution to fit the present scenario. One of CBR's essential advantages is that it is easy to follow and understand the inference process they conduct, which has prompted its use in, for example, the medical domain [10], [11].

* Corresponding author.
E-mail addresses: eamador@fi.upm.es (E. Amador-Domínguez), emilioserra@fi.upm.es (E. Serrano), daniel.manrique@upm.es (D. Manrique), jbajo@fi.upm.es (J. Bajo).

This paper proposes a hybrid CBR-deep learning model to tackle the problem of radiology report writing assistance. The main efforts in the radiology domain reside within image-related tasks, such as diagnosis or X-ray image segmentation. In this image-dominated field, medical reports play a secondary role, mostly used to support the aforementioned tasks. Thus, high quality labelled textual data in this domain may not always be available, which hinders the use of deep learning techniques.

The proposed approach uses a CBR model to work with a few cases that can scale up, assisted by deep learning models to improve its performance. Therefore, it is a blended solution between a knowledge-based system [12], where the knowledge must be elicited, and a deep learning model, where no expert assistance is required. The proposed CBR model considers expert knowledge as an input to improve and validate the stored cases, but it does not rely exclusively on this knowledge to function.

This framework has been developed under a Horizon 2020 research project AI4EU [13], whose goal is to provide users with artificial intelligence resources that satisfy specific user necessities. Moreover, resources developed under this project should be explainable, verifiable, physical, collaborative, and integrative [14]. The proposed system meets all these specifications, as the usage of a CBR model ensures explainability, collaboration, and verification. The combination of different machine learning modules within the proposed model enables integration. Simultaneously, the introduction of sensor-retrieved and human-generated data ascertains physical interaction between

Please cite this article in press as:
E. Amador-Domínguez, E. Serrano, D. Manrique, J. Bajo. A Case-Based Reasoning Model Powered by Deep Learning for Radiology Report Recommendation, International Journal of Interactive Multimedia and Artificial Intelligence, (2021), http://dx.doi.org/10.9781/ijimai.2021.08.011

the users and the framework. The implementation of the proposed framework for the radiology domain is available as a resource in the project platform with an open-source software license [15]. This implementation can be accessed by any user and modified accordingly to fit different purposes and work domains.

The remainder of the paper is organized as follows. Section II provides an insight into the related works. Section III presents the proposed hybrid CBR-deep learning model for radiology report recommendation, while the architecture and implementation of the model are explained in Section IV. Section V reports experimentation and obtained results. Finally, Section VI draws conclusions and future work.

II. Related Works

Case-Based Reasoning is widely used in the medical domain due to its adaptability and interpretability. CBR models have been successfully employed for diagnosis [10], [16], medical decision support [17], and patient monitorization [18], amongst other tasks.

CBR methodology [19] implements a continuous cycle, where the model improves over time by assimilating the knowledge acquired from the resolution of previous problems or cases. Subsequently, the model's performance relies on the number of stored cases and the relevance of those cases concerning the given situation. Mechanisms to efficiently store and manage the acquired knowledge are needed to reach an optimal case set. Several works have explored these issues, presenting new approaches for case retrieval and case-based maintenance.

Qin et al. [20] use heuristics to develop a new and efficient case retrieval algorithm. Daengdej et al. [21] study the substitution of the standard distance-based retrieval algorithm by a statistic-based method, focusing on the automobilist sector. Regarding case-based maintenance, Torrent-Fontbona et al. [11] present a model that combines case-based redundancy reduction with weight attribute learning to store and manage the cases efficiently. Nasiri et al. [22] explore the introduction of ontologies to manage and ensure the stored cases' semantic consistency.

Opposite to these naïve approaches, recent proposals aim to integrate deep learning techniques within the CBR cycle. As previously stated, while deep learning models are currently state of the art in most benchmarking tasks, their lack of explainability hinders their usage in the medical domain. Nonetheless, CBR methodologies can benefit from deep learning qualities by integrating them into different parts of the cycle. Such is the case of the work by Marie et al. [23], where they combine a CBR model with a Convolutional Neural Network to segment kidney radiographs. This proposal presents CBR as a solution to quality data insufficiency, serving as a preprocessing and augmentation mechanism for the network. Similarly, Corbat et al. [24] employ a combination of CBR with deep learning to efficiently segment medical images. Finally, Lamy et al. [25] study the possibility of exploiting CBR models' interpretability to explain the predictions of a deep neural network over a breast cancer dataset.

Other proposals focus on the introduction and management of ad-hoc captured data via sensors. The introduction of this data enables the development of several healthcare-related applications. Tang et al. [26] employ a CBR model to analyze sensor retrieved data from nursing homes to develop personalized healthcare plans for the patients. On the other hand, approaches such as Massie et al. [27] and Forbes [28] focus on patient monitorization and risk prevention, detecting potentially dangerous cases.

While Case-Based Reasoning models have been successfully employed for image-related tasks, including the radiology domain, their applications on textual data have been much less explored. Deep learning techniques are currently state of the art in most radiology-related tasks, such as medical text classification [29]–[31], diagnosis [32]–[34] and event detection [35], [36]. Some works explore the idea of assisting experts in the generation of medical reports. Toledo et al. [37] propose a prototype of web-based speech recognition for the construction of medical reports, while Donnelly et al. [38] evaluate the comparison between radiology free-text versus structured reports.

III. Deep Learning Supported Case-Based Reasoning for the Generation of Medical Reports

This work presents a CBR deep learning supported model to assist in the medical report generation task. The proposed framework does not automatically generate medical reports but serves as an assistant that provides formal corrections, references, and suggestions. Opposite to the methods studied in Section II, the case-based reasoning model is the core of the proposal. Besides, the user is actively involved in the system's learning procedure, determining which outputs are valid and not, directly impacting the learning process.

The proposed case-based reasoning framework comprises four stages in a cycle: retrieve, reuse, retain, and revise. Fig. 1 presents an overview of the model, showing its four cyclic phases, the interactions between them and the case set, and between the system and the user. The design of the framework is modular to make it easily customizable to fit different problems and domains. The figure depicts interchangeable elements as building blocks.

A. Retrieve

The cycle begins when the user introduces a new problem or case. A case can be either a simple draft of a medical report, or include additional information such as images, specific terms, or references. When the user introduces a new case into the system, the first step is to determine the closest ones from the existing case set. A naïve approach to this issue is to use a simple K-NN search, where the amount of desired cases to retrieve, K, is set, and the selection is purely based on distance criteria between samples. While this approach offers a straightforward, efficient solution, two main shortcomings hinder its usage for the proposed system. First, the input data is not measurable. Second, the usage domain is expert-oriented, so that it requires more specific, hand-crafted criteria to retrieve similar cases accurately.

While the similarity between medical reports can be measured according to specific metrics like the age of the patient or demographic data, there are no fixed, static criteria that enable direct comparison. Moreover, while some elements may remain stable between comparisons, some criteria may vary between users. The proposed framework includes an indicator-based retrieval algorithm to tackle this issue. Instead of comparing each case as a whole, the algorithm evaluates four different indicators per case. The four considered indicators I1, I2, I3, and I4 are:

- I1: *Image Comparison*. While images may be irrelevant in some medical areas, they are the cornerstone in others like neurology or dermatology. In such fields, pictures provide essential information that should not be diluted within the text but treated separately. Several methods can be considered for image comparison, ranging from histogram to feature vector comparison. While Convolutional Neural Networks are possibly the most robust way to represent images in a fixed dimensional space, some simpler alternatives can be considered for the task. Feature matching algorithms such as SURF [39], ORB [40], or KAZE [41] offer interpretable, easy to implement options. Nonetheless, these algorithms are quite sensitive to potential image failures such as light flashings and cannot capture finer-grained information. A possible solution

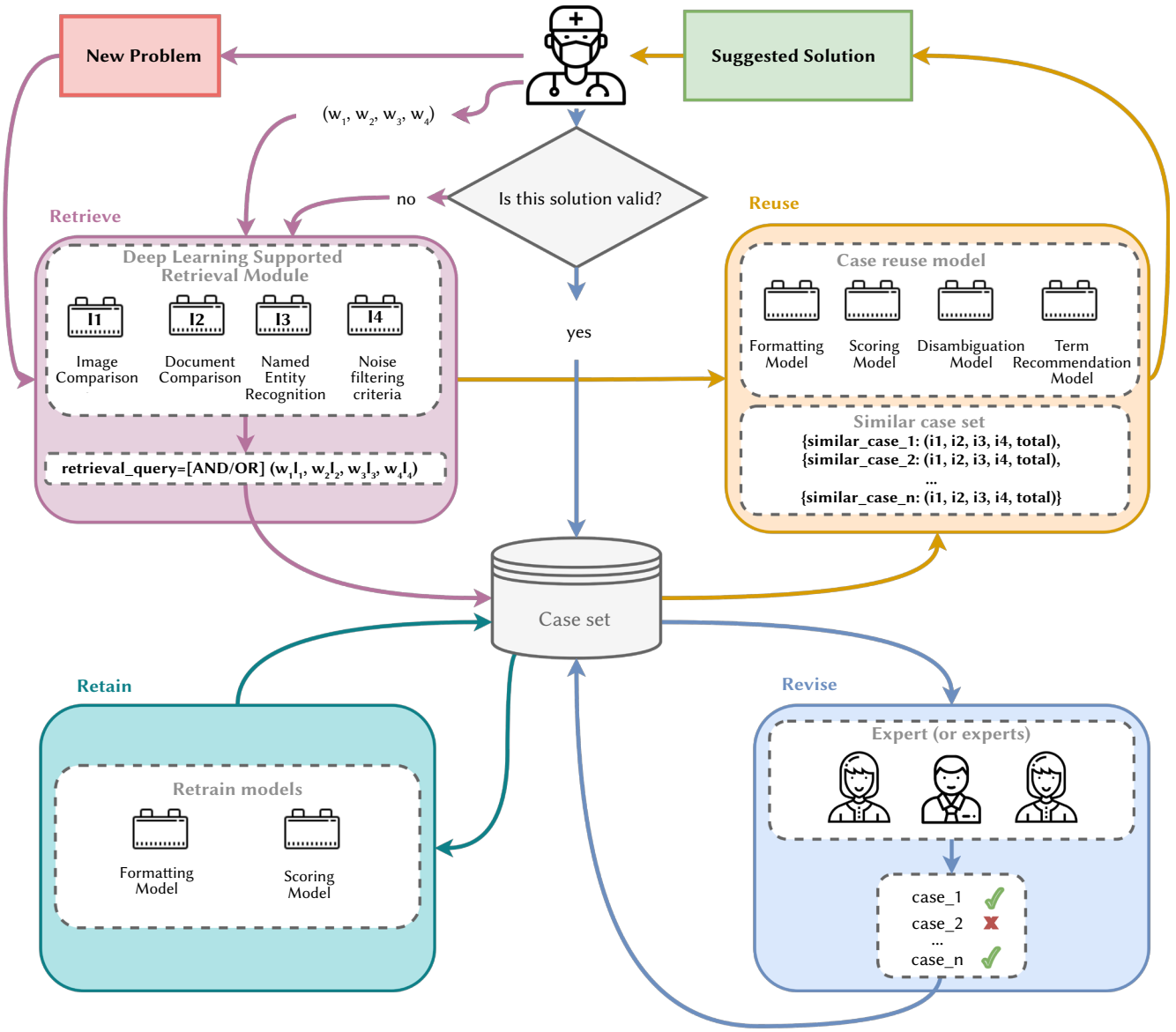


Fig. 1. An Overview of the Case-Based Reasoning System proposed.

to this issue is to combine differently generated feature vector representations into a unique vector. Once the image is embedded into a vector, a distance-based comparison can be established to ascertain the similarity between images.

- I2: *Document Comparison.* As in the case of images, several approaches can be considered to establish similarities between documents. While some non-feature-based methods can perform this task, their performance is entirely lacking compared to those where documents are embedded into a vector space and compared using different distance-based approaches. Models such as Word2Vec [42] or BERT [43] are the preferred choices for document representation, but more straightforward methods such as bag-of-words or TF-IDF can also be employed. However, these models cannot capture underlying semantic information, leading to less expressive representations at a faster cost. Regarding comparison, multiple methods can be considered depending on both the type of documents and the purpose. In this respect, cosine similarity, word's mover distance [44], or probabilistic based methods, which convert the embedding into a probabilistic distribution before comparison, are suitable choices.
- I3: *Named Entity Comparison.* Named Entity Recognition, or NER, is one of the main natural language processing tasks, particularly significant in the medical domain. In this task, the goal is to detect and label relevant terms within the text, such as people, places, or dates. While its usage is extended to a wide range of domains, there is a particular interest in developing NER models that focus specifically on detecting medical-related terms such as disease names, proteins, or drugs. Examples of clinical NER models are CliNER [45], BioBERT [46], or SciBERT [47]. Discovering relevant labelled terms within the documents is a way to detect and retrieve related documents. Therefore, only those cases whose reports contain user-specified terms will be considered for retrieval, reducing the search scope.
- I4: *Noise Filtering Criteria.* In addition to the previous indicators, additional filtering criteria may be specified to discard unfitting cases. They regard formatting specifications, like the absence of images or language employed, or type of content such as unidentified words like typos or abbreviations. Filtering criteria can be as restrictive as required.

For each of these four indicators, the user can establish a threshold value. Indicators can be combined either in a conjunctive or disjunctive

way, depending on the user's goal. These criteria are then translated into a search query, which will then be used to retrieve the top N, or all, existing cases meeting the user-provided constraints.

```
query = { I1 >=0.85 , I2 >=0.7 , I3=['
    Pulmonary Disease ', 'Pneumonia '], I4
    =[lang= 'en ', identified_abbrv_rate >=
     0.9 ], N=5, operation= 'OR '}
```

Listing 1. Example of a retrieval query.

Listing 1 depicts an example of a retrieval query. According to this query, the system provides the user with the top 5 cases that either:

- Include images that are at least 85% similar to the given ones.
- Contain a clinical report that has a similarity of at least 70%.
- Contain the medical terms 'pulmonary disease' and 'pneumonia'.
- Are written in English, and at least 90% of the report's abbreviations have been disambiguated.

Cases that meet the retrieval criteria are ordered in decreasing order according to their cumulative similarity across the four indicators. Then, the top *N* cases demanded by the user are returned.

B. Reuse

Once the user has defined the retrieval criteria, the existing cases that fit the imposed constraints are selected and presented. A brief explanation of why each case has been chosen is also provided to maintain the system expressive and understandable. Providing information such as the similarity rates between the current case and the retrieved ones explains the system's decision process while giving further guidance to the user. From the instances retrieved in the previous stage, several operations are performed to obtain precise, expressive information that will aid the user in the report generation task. Fig. 1 shows four different modules in this stage to provide information to the user:

- *Formatting Module.* Readability is one of the most desirable features when it comes to any written report. It encompasses content matters, such as syntactic cohesion, and more straightforward format issues like proper paragraph and sectioning when required. In medical documents, while there may be differences from one domain to another, there is generally a fixed, basic structure to present the data. On a general view, a medical report comprises four main sections:
  - *Indication.* A brief introduction to the case, giving superficial information about the patient and the observed symptoms.
  - *Comparison.* References to previous existing reports of the given patient.
  - *Findings.* In-depth information about the potential causes of the symptoms, as well as additional observations about the patient.
  - *Impression.* Conclusions and diagnosis.

  A formatting module is included in this stage such that when a report in raw format is introduced, it can be adequately divided into paragraphs and sections.
- *Disambiguation Module.* Abbreviations are quite usual in the medical domain due to the existence of a high amount of complicated compound term names. However, while most of these abbreviations may be universal and easily understandable by any professional, some can still be obscure for a regular reader. A potential solution for this issue is to include a module that not only detects the abbreviations contained in the report but offers disambiguation suggestions for them. While this may extend the document, it also highly improves its readability, as it removes any potential misunderstandings induced by the abbreviations.
- *Term Recommendation Module.* As previously stated, Named Entity Recognition is particularly prominent in the medical domain. These models can accurately detect relevant terms and group them in a fixed set of given categories. These categories are usually related to each other in some manner, and, subsequently, so are the corresponding terms. For example, given a report about a patient with pulmonary disease, terms such as *(pneumonia, disease)* and *(chest x-ray, test)* may appear together frequently. This module offers these correlated terms to the user as suggestions. To obtain these suggested terms, named entity recognition is performed over the previously retrieved relevant cases, receiving a set of terms with their corresponding category. This set of terms are then flattened, cleaned, and presented to the user in their corresponding categories. Hence, if the user is writing a report containing the word *pneumonia*, but does not include *chest x-ray*, the system may recommend the inclusion of this term, as this correlation has previously appeared in those cases detected as related.
- *Scoring Module.* Finally, the system presents the user with a validity score, indicating whether the report, in its current form, is readable and understandable enough. This score can vary from a simple binary value (valid or invalid) to a star-scored base method, to a finer decimal system.

It is important to note that the recommendations and suggestions offered by the system are not final. The user must decide which of the given suggestions are to be applied to the current report.

Once the report's state satisfies the user, the system generates a new case and stores it into the case base. It also presents the original document as the problem and the final state as the solution. New cases are marked as *pending validation* and will not be added to the case base until the experts validate them.

C. Revise

Once the report satisfies the user, after applying any or none of the suggestions provided, the system generates a new case. However, it cannot be added directly to the case set as it may include errors that can hinder the system from improving. Moreover, if the system stored unreliable cases without any revision, they might be presented to the next users as solutions, misguiding them. Therefore, an intermediate step is required to ensure that those instances included in the case set are useful and needed.

A panel of experts must perform this task, manually checking *pending-on-validation* cases to provide them with a coherent score with the criteria implemented in the scoring module. Hence, if the system offers the user a binary score, the experts must also grade the cases following this criterium. Experts can also modify or correct minor mistakes within the cases before validating them to ensure their quality.

D. Retain

As noted in Section II, one of the biggest concerns regarding case-based reasoning systems is how to handle the ever-growing number of cases. Ideally, the case base should be composed of an optimal number of instances where the problem coverage is maximum, while the number of cases is minimum. However, while infeasible cases may not help the user, they improve the scoring models' accuracy. For this purpose, invalidated cases are also stored separately from the case base, where they can be recovered when necessary.

New cases are being regularly introduced into the case base and, subsequently, they must affect the system's behaviour. CBR models nurture themselves by adding further information, which keeps

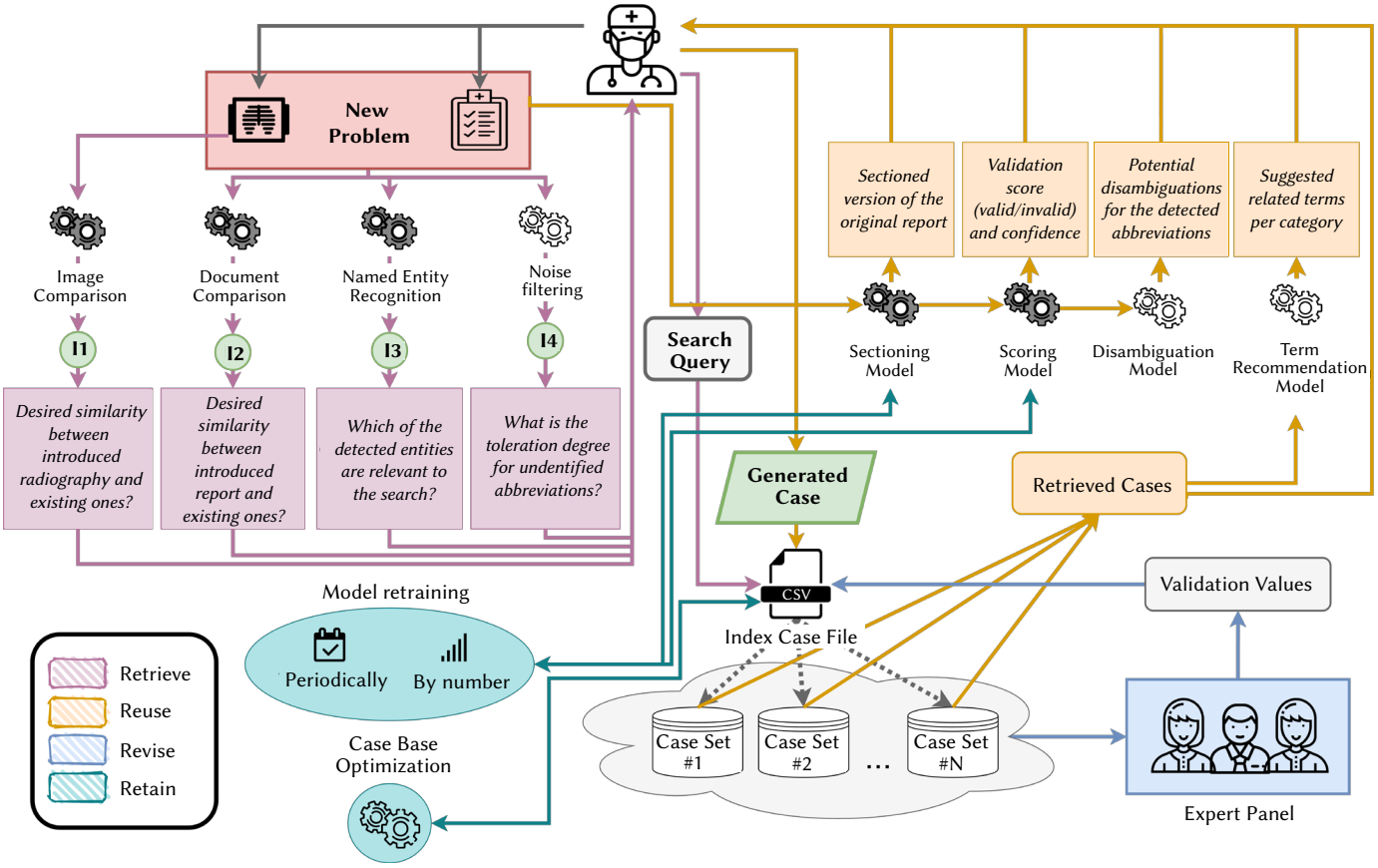


Fig. 2. Data flow of the proposed implementation. Coloured elements depict each stage of the CBR cycle. Solid gears represent deep learning powered modules, while clear ones represent non-machine learning modules.

them updated and usable throughout time. Aside from case-based maintenance, module updates also are conducted in this stage. These updates can be either a replacement, such as switching from regular expressions to machine learning models, or just a retrain of an existing model. Updates can be either scheduled periodically or when a particular milestone of case numbers is reached. The scoring model can eventually substitute the panel of experts once it has gained enough maturity.

IV. System Architecture and Implementation

An implementation and case study is provided to illustrate the proposed framework. In this case study, the system focuses on the treatment and generation of radiology reports. This context presents a challenging scenario where both images and textual information are highly relevant to the problem. The implemented resource instantiates the proposal depicted in Fig. 1, selecting the appropriate paradigms for each of the eligible modules. Fig. 2 illustrates the data flow of the system.

The framework implements a four-layered software architecture. Before defining the CBR, some issues need to be addressed, such as data management and storage mechanisms. An indexed storage model is employed to deal with the ever-growing nature of the case set while still enabling fast retrieval. In the proposed storage system, cases are stored either in a distributed or centralized way and are referenced in an index file. The index file contains each case's location and its respective retrieval indicators to accelerate the retrieval process. Before starting the CBR cycle, preprocessing operations may be required to fit the system's constraints, such as separating images from text or formatting the report.

In the context of radiology, a case comprises a radiograph and a brief text summarizing the most relevant findings of the image, alongside additional information about the patient. While these two elements are enough to define a new problem, the user can also provide further information, as depicted in Fig. 3.

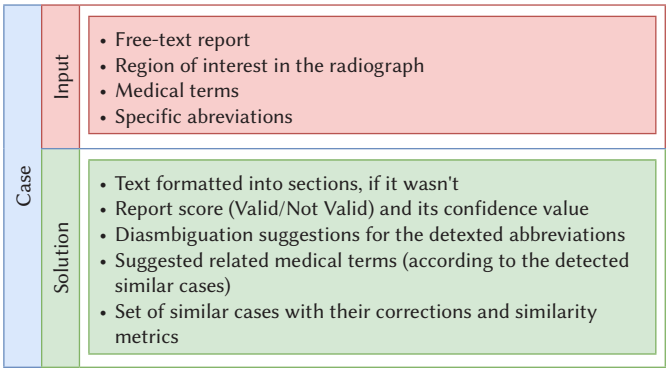


Fig. 3. Case composition of the provided implementation.

The retrieve stage begins once the user introduces a new problem into the system. Then, case indicators are then computed as follows:

- I1. *Image comparison:* In the current domain, images are black and white radiographs. Hence, there is not much variation between samples. A convolutional neural network generates the embeddings to capture the subtle differences between radiographs and enable an accurate comparison. A white-box feature detection algorithm is also employed to add a supplementary explainable level to the comparison. KAZE [41] generates fixed dimension descriptors from the key points detected in an image. These key points can

be indicated in the picture, providing a visual explanation based on which the comparison is performed. KAZE representation is averaged with that obtained from the convolutional neural network. Then, it generates a unique embedding that combines both interpretable and abstract knowledge. The comparison is performed based on this final combined embedding.

1. I2. *Report comparison:* This task employs a pretrained NLP model specific to clinical data. This model provides single word embeddings for each of the tokens within the text, sentence-level embeddings, and document embeddings. The latest type is used to generate comparable report representations.
2. I3. *Named Entity Recognition:* This task uses CliNER [45]. This framework provides a series of models trained over a sizeable clinical corpus, capable of identifying the following entity types: diseases, treatments, and tests. As mentioned in Section III, multiple NER choices in the clinical domain range from fine-grained information, such as protein detection, to general type identification such as drugs versus diseases. CliNER offers an intermediate solution that fits the present scenario.
3. I4. *Noise filtering:* The same NLP model employed for report comparison is used to filter noise. In this context, noise refers to those elements on the text that can not be identified as tokens, and therefore they have no embedding nor meaning attached. The report is run through the NLP model to detect these conflicting terms, obtaining a set of identified tokens. Noise is then calculated as the proportion of identified tokens concerning the total amount of elements contained within the text.

These indicators are only computed once per case and are stored in the index file to accelerate the retrieval process. The user is then asked to specify which threshold values are considered for each of the proposed metrics, how to combine the indicators (conjunctively or disjunctively), and the number of related cases k which must be retrieved. Fig. 2 depicts a descriptive representation of the values inquired to the user, represented by purple-coloured boxes, where the threshold value for each indicator is posed as a human-readable question. For example, in the case of I2 (document processing), the framework would ask the user 'what is the minimum similarity acceptable between the current and the existing reports?'. These queries must be clearly presented and understandable to the user, as the success of the retrieval phase is directly related to the constructed query.

Once the search query is formulated, a comparison between the current problem and the existing cases is performed. Instead of retrieving each complete case individually from the case set, the comparison is performed based on the case indicators contained in the index file. Thus, when an existing case is detected as fitting, its full content is retrieved from the case set. A summary of each indicator's similarity metrics is attached and presented to the user alongside the case itself.

The retrieved cases are then used as a support for the term recommendation module This list containing the retrieved, top k similar cases is also provided to the user. Orange-coloured boxes in Fig. 2 present the different stages of the reuse phase. As shown, the named entities identified in the retrieved cases are processed by the term recommendation module, which groups the detected terms according to their type. Duplicate entries are also removed. The resulting term aggregations are then presented to the user, providing guidance on which entities could be related to the ones detected in the current case. Additionally, as depicted in Fig. 3, the following content and format suggestions are provided to the user as part of the solution:

- *Sectioned version of the report:* A bi-directional long-short term memory is employed for the formatting task. The problem itself is treated as a classification problem, where each sentence is labelled according to the section where it appeared. The goal of the model is to predict the best fitting section for each sentence. When formatting a new report, sentences are presented in the same order they are listed in the text to avoid permutations in the content.
- *Potential disambiguations for the detected abbreviations:* Similarly to the noise filtering operation, a set of unidentified tokens within the text is first obtained. The elements in this set are then looked up in the medical terminology SNOMED-CT, bringing the best applicable medical term for the input abbreviation.
- *Case validation score and confidence:* Binary scoring is employed in this implementation, categorizing the cases between valid and invalid. While a case is only validated or discarded in the revising stage, this score informs the user of whether the current state of the report would be considered appropriate or not. For this task, a random forest is used.
- *Suggested related terms per category:* Named entity recognition is applied to the content of the top N retrieved cases, obtaining a set of *(term, category)* tuples. Duplicates are removed from the set. These terms are then presented to the user grouped by category. CliNER [45] identifies named entities within the report, categorizing the detected terms into three types: disease, test, and treatment.

The system presents these suggestions to the user, who can freely decide which must be applied to the current problem. Once the appropriate modifications over the original report are performed, the generated solution is stored alongside the initial problem, comprising a new case. New cases are labelled as *pending on validation* and will not be shown to future users until experts have reviewed them.

During the phase of revise (depicted in Fig. 2 in blue-coloured boxes), an expert panel is in charge of regularly validating the pending cases, deciding which are valid and should be presented to the users and which are not. The validation status of each case is also referenced in the case index file to ease the filtering of which cases should be shown. Commonly, invalid cases are deleted from the case set, as they intuitively do not provide valuable information to the user. However, these cases are necessary to train and obtain robust scoring models that may even replace the expert at some point. Corrupted cases can be exploited for the benefit of the system, improving its performance.

Once there are enough classified cases, the retain stage begins, as depicted by the green-coloured bubbles in Fig. 2. In this stage, both the scoring and sectioning models employed in the reuse phase are retrained using the case set's information. Models can be retrained following either a periodical or a quantitative approach. Periodical retraining ensures that the model is kept updated and improves the final quality of the results. However, this approach presents a shortcoming: when there is a limited number of cases in the case set, the model's generalization capability will be logically limited. Additionally, case base optimization is performed in this stage. As previously stated, one of the biggest challenges in CBR models is to devise a management protocol for dealing with the ever-growing amount of cases. In the proposed framework, case base optimization is performed by maximizing case relation. First, a global linking process is launched amongst cases, computing the top 5 most similar cases per instance. Cases that are listed as related by at least one different case are kept in the case base. Unreferenced cases are removed from the case base, thus not shown to the users, but are still considered for model training.

V. Experimentation and Results

Experimentation based on the proposed implementation is set up to assess the performance and accuracy of the proposal. The majority of the studied approaches focus on evaluating the retrieval

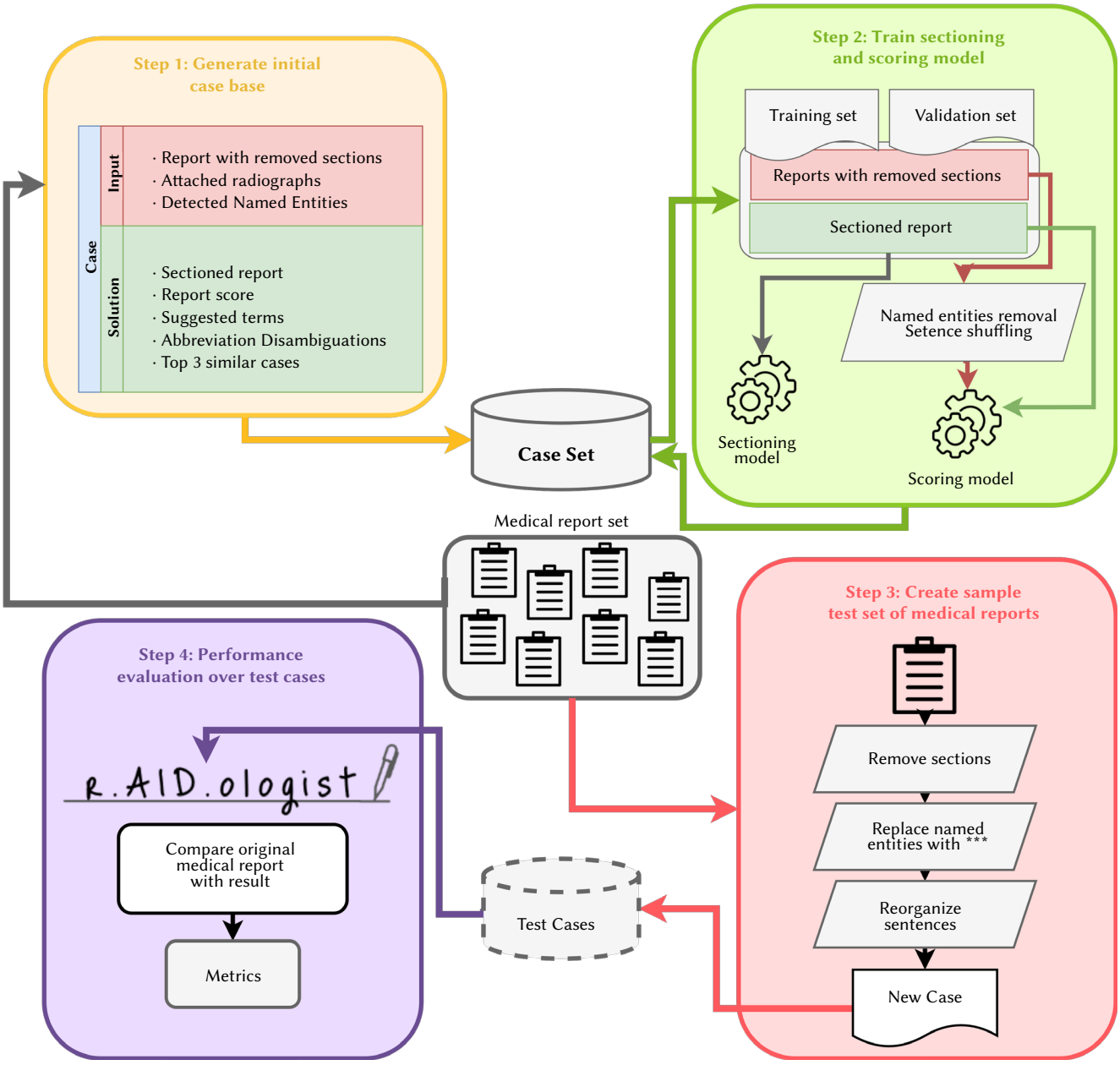
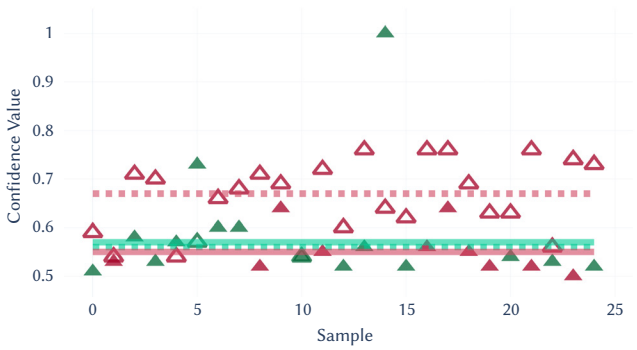


Fig. 4. Overview of the experimentation process conducted to evaluate the system.

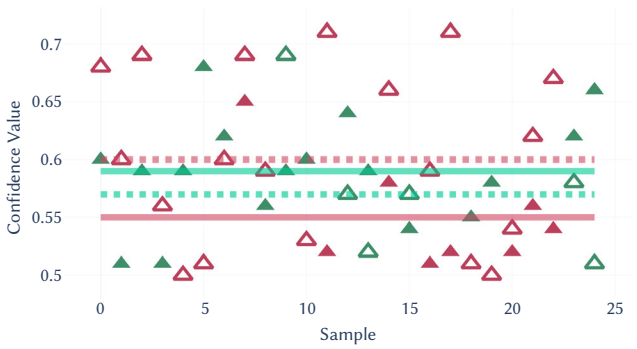
strategy, as it is a crucial element of CBR systems. Our proposal, however, relies on user-input queries to retrieve the most fitting cases. Hence, assessing the system performance based solely on the retrieval approach would not be representative enough, as the success of this stage is directly related to the user criteria.

Since the considered context is highly expert-oriented, it is not trivial to perform a quality assessment of the framework without expert information assistance. Therefore, an alternative evaluation approach capable of quantitatively measuring the performance of the model is required. The proposed evaluation procedure assesses the performance of the proposal for the report correction task. Fig. 4 depicts the conducted evaluation process, comprised of the following stages:

1. *Step 1: Generate the initial case base.* As previously stated, the case base is at the core of any case-based reasoning model. In this first step, a set of medical reports is converted into cases, composing the initial case base. Out of all the available medical reports, a sub-sample of 25 elements is randomly selected to be later used for testing. These randomly selected elements are not included in the case base. From the remaining cases, each medical report is stripped, when possible, from its sections, creating the input of the case. If a list of named entities and abbreviations are provided for the report, they are also included as the input. If the original report was already sectioned, its content is stored in the case solution as a sectioned report. The remaining solution values (score, suggested terms, and similar cases) are updated in the following step.
2. *Step 2: Train sectioning and scoring model.* At this stage, the cases contained in the case base only include the input (the original report stripped of its sections) and its corresponding solution (the original report without any modifications). These are the only attributes required to train both the sectioning and scoring model. The existing cases are randomly divided into two sets: training and validation. As stated in Section IV, sentence-based classification using a bi-directional long-short term memory is used to section each report. The sectioning model is trained using the case solution, where each report is split into sentences,

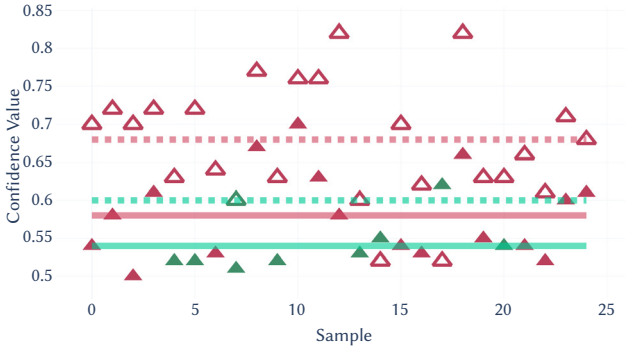


(a) ECGEN 50 case set

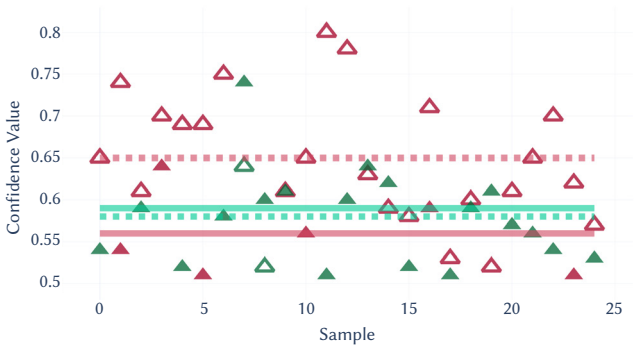


(b) MIMIC-CXR 50 case set

Fig. 5. Validation status on the 50-case-set for each dataset. Empty triangles denote the state of the case before the system applies the appropriate corrections. Solid triangles indicate their status afterward. Green and red colors depict valid and invalid cases, respectively. The y-axis represents the confidence value assigned by the system to the validation score.Average values are depicted as horizontal lines: discontinuous and continuous lines represent before and after values, respectively. The colors employed for the cases match the average lines.



(a) ECGEN optimal case set



(b) MIMIC-CXR optimal case set

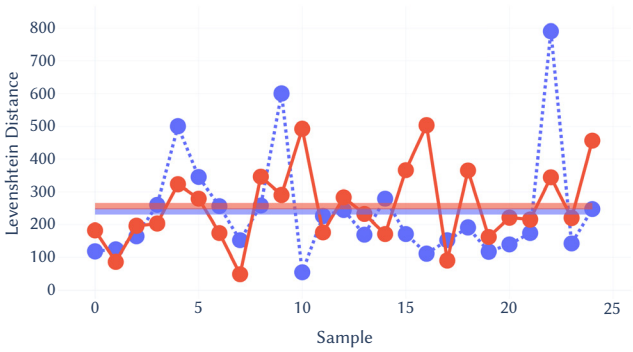
Fig. 6. Validation status on the optimal case base for each dataset. Empty triangles denote the state of the case before the system applies the appropriate corrections, while solid triangles denote their status afterward. Green and red colors depict valid and invalid cases, respectively. The y-axis represents the confidence value assigned by the system to the validation score. Average values are shown as horizontal lines: discontinuous and continuous lines illustrate before and after values, respectively. The same color code employed for the cases is used for the average lines.

and each sentence is labelled with the value of its corresponding section. For the scoring model, both the input and the solution of each case are required as this model feeds positive and negative samples. Therefore, case inputs comprise the negative sample set, while solutions comprise the positive sample set. A sequence of escape characters substitutes the named entities on each non-sectioned report, and the sentences are randomly reordered to further corrupt the negative samples. These sets are then used to train a random forest classifier, which acts as the scoring model. Once both sectioning and scoring models have been trained and validated, the case base is updated, adding each report score. Named entities, disambiguations, and similar cases are also updated.

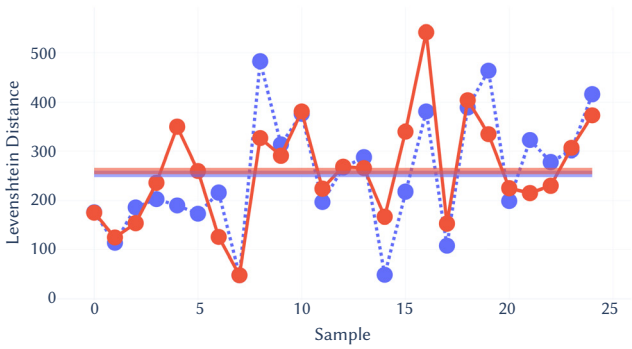
3. *Step 3: Create a sample test set.* A set of input cases is created from the medical reports set aside for testing in Step 1. A comparison between the provided solution for a corrupted version of the input versus the original report is conducted to assess the proposal performance. Therefore, for each element in the test set, the following corruption operations are performed to create an input case: section removal, named entity replacement by a character sequence, and sentence reordering.
4. *Step 4: Performance evaluation over the test set.* The generated inputs are then passed onto the system, which attempts to provide a valid solution for the input permuted report. Alongside the corrected report, the framework presents a list of recommended terms and disambiguation abbreviations. The corrected version of the report is then compared with the original. The model should reorganize the sentences into sections in a cohesive order and suggest introducing the named entities previously stripped from the report. The following metrics are computed to assess the framework performance:
   (a) The validation score provided by the model before and after the corrections.
   (b) The Levenshtein distance between the original report and the suggested correction.
   (c) The proportion of entities detected on the original report pointed out by the model.

Two different radiology datasets are considered for evaluation: MIMIC-CXR [48] and Open-I's radiology set, denoted as ECGEN [49]. MIMIC-CXR contains complete medical reports in plain text format, without any additional information. On the contrary, Open-I provides both images and named entities alongside the medical report, and additional metadata. From each dataset, two initial case bases are generated, composed of 50 and 200 cases, respectively. Cases are generated from a random sampling of reports from each considered dataset. The developed implementation is used to conduct the experimentation.

The initial 50-element case base serves as a baseline to assess the performance of the framework when the number of cases is limited. Applying the retain criteria in this scenario may not have any impact, as most or all cases may be related between them. In the initial

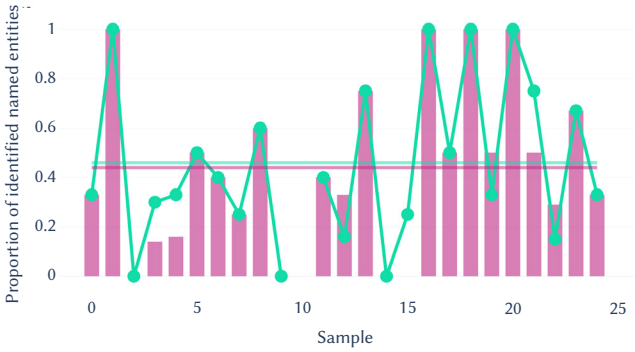


(a) ECGEN case set

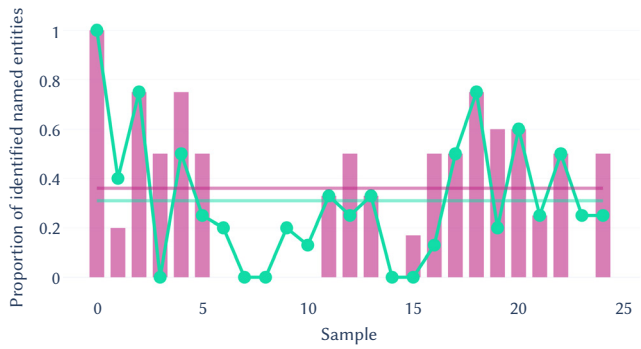


(b) MIMIC-CXR case set

Fig. 7. Levenshtein distance per sample on each studied dataset. Purple and orange lines depict the results obtained on the 50-case and optimal set, respectively. The horizontal lines represent the average values, using the same color code employed for the results.



(a) ECGEN case set



(b) MIMIC-CXR case set

Fig. 8. Proportion of named entities correctly identified on each studied dataset. Purple bars depict the results obtained for the 50-case set, while green lines illustrate the results achieved in the optimal case base. The horizontal lines represent the average proportions, using the same color code employed for the results.

200-element case base, where the amount of existing elements is four times the size of the prior case base, retain criteria can be successfully applied, obtaining the optimal case base. The resulting optimal case bases are comprised of a total of 187 elements for MIMIC-CXR and 90 cases for ECGEN. Two different case bases are considered per dataset: a baseline 50-element case base, and an optimal case base.

Fig. 5 depicts the results obtained by the model when the case base comprises only 50 cases. Despite the simplicity of the case base, the framework still offers noteworthy results, accurately correcting most of the initially corrupted cases. This performance can be clearly observed in the results obtained in the ECGEN dataset (Fig. 5(a)), where most of the initial cases are noted as corrupted with a high confidence value and turn into valid after the corrections applied by the system. In the case of MIMIC-CXR, this improvement is not as noticeable as some cases remain considered invalid by the system after the corrections. However, as illustrated by Fig. 5(b), even in those cases still denoted as invalid after the modifications, the confidence value assigned by the system dramatically diminishes. This decrement evidence that, even though the report is still marked as invalid, the system corrections significantly reduce the corruption level of the report.

Using the optimal case set of each studied dataset impacts positively the performance of the model, as shown in Fig. 6. In this optimized context, the results are slightly more polarized than in the previous case, and most of the original corrupted cases are corrected and validated once processed by the system. Moreover, the confidence levels are higher than in the 50-case set, indicating that the framework can train more refined models, better distinguishing between valid and corrupt cases. Furthermore, considering the optimal case set for each particular dataset soothes the existing differences in performance. While in the 50-case set, the results obtained on the ECGEN dataset were slightly better since more reports were correctly modified and denoted as valid, in the MIMIC dataset the reports underwent a correction process that was insufficient to validate the case.

Levenshtein distance [50] between each original and corrected report pair is also computed to further assess the correction capabilities of the model. While different text similarity metrics could be considered for evaluation, such as cosine similarity or Jaccard index, these metrics do not consider text order. As previously stated, test cases are generated by stripping sections, permuting sentence order, and removing named entities. Therefore, even after the corruption process, both the original and corrupted report are almost equal in terms of content. Thus, an order-sensitive metric is required to ascertain the similarity degree between the original and corrected report. Fig. 7 illustrates the Levenshtein distance per pair of an original and corrected report on each studied dataset and case base. As shown in Fig. 7(a), Levenshtein distances in ECGEN, on both 50-element and optimal case bases, remains fairly similar throughout cases. A similar occurrence happens in MIMIC-CXR cases (Fig. 7(b)), where the distance between original and corrected reports remains akin. While finding the optimal case base benefited the framework results in the validation scenario, this improvement is not reflected regarding report sectioning and reordering. This flaw may be solved with the introduction of user input. While corrupted samples have been artificially generated from simple text editing operations in this experimentation, user-corrected reports may be more expressive and richer in content, leading the model to identify more complex correction patterns that would subsequently lead to better results.

The amount of named entities correctly suggested by the system is also provided, illustrated in Fig. 8. As stated in step 4 of the

experimentation process, named entities in the original report are substituted by escape characters as part of the corruption process. Figures 8(a) and 8(b) depict the proportion of named entities stripped from the original report and correctly suggested by the framework on each dataset. The results show that, when the case base is optimized, the amount of detected entities either improves or holds. This is particularly noticeable in ECGEN's results (Fig. 8(a)). Only in three cases, the amount of detected entities slightly decreases with the optimized case base, but significantly improves in four other cases. In MIMIC-CXR (Fig. 8(b)), the results are not as consistent, which could be due to the difference in the case base size between both studied datasets. MIMIC-CXR has double the cases on its optimized version than ECGEN. Named entities are suggested based entirely on the top k most similar cases identified by the system. Hence, if the retrieved similar cases contain few named entities, this would directly impact the number of suggestions provided by the system. A way to overcome this issue is to increase the value of k.

VI. Conclusions and Future Work

This work presents a hybrid framework that combines a case-based reasoning system with several deep learning models to help health professionals generate medical reports. The proposed system is fully modular, making it effortlessly adaptable to several scenarios and heterogeneous data. A use case focusing on the development of radiology reports is provided to illustrate the proposal. An open-source implementation for this particular use case named r.AID. ologist is provided under the AI4EU platform. This implementation is used to assess the performance of the proposed framework. Two different radiology datasets are used: MIMIC-CXR and ECGEN. For each studied dataset, two different scenarios are considered: a baseline 50-element case base and an optimized case base. The results show that, even without external user validation, the system considerably benefits from optimizing the case base, as it increments its sensibility. Moreover, the results also evidence the robustness of the proposal even when the amount of available information is minimal, being capable of properly correct formatting errors while providing relevant suggestions, such as related terms or abbreviation disambiguations.

Acknowledgments

The authors thank the reviewers and editors for their valuable comments and suggestions, which have improved this paper. This project has received funding from the Horizon 2020 research and innovation programme of the European Union, under grant agreement No. 825619. This work has also been supported by the Autonomous Region of Madrid through the program CABAHLA-CM (GA No. P2018/TCS-4423) and by the "Universidad Politécnica de Madrid" under the program "Ayudas para Contratos Predoctorales para la Realización del Doctorado". The authors would like to thank Jérémy Clech and Guillaume Martial from NEHS DIGITAL for their support and numerous comments during the development of this work.

References

[1] Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao,T. Zhang, S. Gao, J. Liu, "Ce-net: Context encoder network for 2d medical image segmentation," *IEEE Transactions on Medical Imaging*, vol. 38, no. 10, pp. 2281–2292, 2019, doi: 10.1109/TMI.2019.2903562.

[2] D. Jha, P. H. Smedsrud, M. A. Riegler, D. Johansen, T. D. Lange, P. Halvorsen, H. D. Johansen, "Resunet++: An advanced architecture for medical image segmentation," in *2019 IEEE International Symposium on Multimedia (ISM)*, San Diego, CA, USA, 2019, pp. 225–2255.

[3] D. Jha, P. H. Smedsrud, M. A. Riegler, P. Halvorsen, T. de Lange, D. Johansen, H. D. Johansen, "Kvasir-seg: A segmented polyp dataset," in *MultiMedia Modeling*, New York, New York, USA, 2020, pp. 451–462, Springer International Publishing.

[4] A. Avetisyan, M. Dahnert, A. Dai, M. Savva, A. X. Chang, M. Niessner, "Scan2cad: Learning cad model alignment in rgb-d scans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, California, USA, June 2019.

[5] B. Yang, S. Wang, A. Markham, N. Trigoni, "Robust attentional aggregation of deep feature sets for multi-view 3d reconstruction," *International Journal of Computer Vision*, vol. 128, pp. 53–73, Jan 2020, doi: 10.1007/s11263-019-01217-w.

[6] J. Liu, Z. Zhang, N. Razavian, "Deep ehr: Chronic disease prediction using medical notes," in *Proceedings of the 3rd Machine Learning for Healthcare Conference*, vol. 85 of *Proceedings of Machine Learning Research*, Palo Alto, California, 17–18 Aug 2018, pp. 440–464, PMLR.

[7] S. Parisot, S. I. Ktena, E. Ferrante, M. Lee, R. Guerrero, B. Glocker, D. Rueckert, "Disease prediction using graph convolutional networks: Application to autism spectrum disorder and alzheimer's disease," *Medical Image Analysis*, vol. 48, p. 117–130, Aug 2018, doi: 10.1016/j.media.2018.06.001.

[8] J. Kolodner, *Case-Based Reasoning*. San Francisco, California, USA: Morgan Kaufmann Publishers Inc., 1993.

[9] M. M. Richter, R. O. Weber, *Case-Based Reasoning: A Textbook*. New York City, New York, USA: Springer Publishing Company, Incorporated, 2013.

[10] G. Costa Silva, E. E. O. Carvalho, W. M. Caminhas, "An artificial immune systems approach to Case-based Reasoning applied to fault detection and diagnosis," *Expert Systems with Applications*, vol. 140, p. 112906, Feb. 2020, doi: 10.1016/j.eswa.2019.112906.

[11] F. Torrent-Fontbona, J. Massana, B. López, "Case-base maintenance of a personalised and adaptive CBR bolus insulin recommender system for type 1 diabetes," *Expert Systems with Applications*, vol. 121, pp. 338–346, May 2019, doi: 10.1016/j.eswa.2018.12.036.

[12] E. Amador-Domínguez, E. Serrano, D. Manrique, J. F. D. Paz, "Prediction and decision-making in intelligent environments supported by knowledge graphs, A systematic review," *Sensors*, vol. 19, no. 8, p. 1774, 2019, doi: 10.3390/s19081774.

[13] "Ai4eu." https://www.ai4eu.eu/. Accessed: 2020-12-21.

[14] "The ai4eu scientific vision." https://www.ai4eu.eu/ai4eu-scientific-vision. Accessed: 2020-12-21.

[15] "Ai4eu." https://www.ai4eu.eu/resource/raidologist. Accessed: 2020-12-21.

[16] M. B. Bentaiba-Lagrid, L. Bouzar-Benlabiod, S. H. Rubin, T. Bouabana-Tebibel, M. R. Hanini, "A case-based reasoning system for supervised classification problems in the medical field," *Expert Systems with Applications*, vol. 150, p. 113335, July 2020, doi: 10.1016/j.eswa.2020.113335.

[17] D. Brown, A. Aldea, R. Harrison, C. Martin, I. Bayley, "Temporal case-based reasoning for type 1 diabetes mellitus bolus insulin decision support," *Artificial Intelligence in Medicine*, vol. 85, pp. 28–42, Apr. 2018, doi: 10.1016/j.artmed.2017.09.007.

[18] E. Lupiani, J. M. Juarez, J. Palma, R. Marin, "Monitoring elderly people at home with temporal Case-Based Reasoning," *Knowledge-Based Systems*, vol. 134, pp. 116–134, oct 2017, doi: 10.1016/j.knosys.2017.07.025.

[19] A. Aamodt, E. Plaza, "Case-based reasoning: Foundational issues, methodological variations, and system approaches," *AI Commun.*, vol. 7, p. 39–59, Mar. 1994, doi: 10.3233/AIC-1994-7104.

[20] Y. Qin, W. Lu, Q. Qi, X. Liu, M. Huang, P. J. Scott, X. Jiang, "Towards an ontology-supported case-based reasoning approach for computer-aided tolerance specification," *Knowledge-Based Systems*, vol. 141, pp. 129–147, Feb. 2018, doi: 10.1016/j.knosys.2017.11.013.

[21] J. Daengdej, D. Lukose, R. Murison, "Using statistical models and case-based reasoning in claims prediction: experience from a real-world problem," *Knowledge-Based Systems*, vol. 12, pp. 239–245, Oct. 1999, doi: 10.1016/S0950-7051(99)00015-5.

[22] S. Nasiri, G. Zahedi, S. Kuntz, M. Fathi, "Knowledge representation and management based on an ontological CBR system for dementia caregiving," *Neurocomputing*, vol. 350, pp. 181–194, jul 2019, doi: 10.1016/j.neucom.2019.04.027.

[23] F. Marie, L. Corbat, Y. Chaussy, T. Delavelle, J. Henriet, J.-C. Lapayre, "Segmentation of deformed kidneys and nephroblastoma using Case-Based Reasoning and Convolutional Neural Network," *Expert Systems*

with Applications, vol. 127, pp. 282–294, Aug. 2019, doi: 10.1016/j.eswa.2019.03.010.

[24] L. Corbat, M. Nauval, J. Henriet, J.-C. Lapayre, "A fusion method based on Deep Learning and Case-Based Reasoning which improves the resulting medical image segmentations," *Expert Systems with Applications*, vol. 147, p. 113200, jun 2020, doi: 10.1016/j.eswa.2020.113200.

[25] J.-B. Lamy, B. Sekar, G. Guezennec, J. Bouaud, B. Séroussi, "Explainable artificial intelligence for breast cancer: A visual case-based reasoning approach," *Artificial Intelligence in Medicine*, vol. 94, pp. 42–53, Mar. 2019, doi: 10.1016/j.artmed.2019.01.001.

[26] V. Tang, K. Choy, G. Ho, H. Lam, Y. Tsang, "An iomt-based geriatric care management system for achieving smart health in nursing homes," *Industrial Management and Data Systems*, vol. 119, no. 8, pp. 1819–1840, 2019, doi: 10.1108/IMDS-01-2019-0024.

[27] S. Massie, G. Forbes, S. Craw, L. Fraser, G. Hamilton, "Fitsense: Employing multi-modal sensors in smart homes to predict falls," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11156 LNAI, pp. 249–263, 2018, doi: 10.1007/978-3-030-01081-2.

[28] G. Forbes, "Employing multi-modal sensors for personalised smart home health monitoring," vol. 2567, 2019, pp. 185–190.

[29] G. Mujtaba, L. Shuib, N. Idris, W. L. Hoo, R. G. Raj, K. Khowaja, K. Shaikh, H. F. Nweke, "Clinical text classification research trends: Systematic literature review and open issues," *Expert Systems with Applications*, vol. 116, pp. 494 – 520, 2019, doi: 10.1016/j.eswa.2018.09.034.

[30] J. T. Oliva, J. L. G. Rosa, "Classification for EEG report generation and epilepsy detection," *Neurocomputing*, vol. 335, pp. 81 – 95, 2019, doi: 10.1016/j.neucom.2019.01.053.

[31] S. Baccianella, A. Esuli, F. Sebastiani, "Variable-constraint classification and quantification of radiology reports under the ACR Index," *Expert Systems with Applications*, vol. 40, no. 9, pp. 3441 – 3449, 2013, doi: 10.1016/j.eswa.2012.12.052.

[32] C. R. Olsen, R. J. Mentz, K. J. Anstrom, D. Page, P. A. Patel, "Clinical applications of machine learning in the diagnosis, classification, and prediction of heart failure," *American Heart Journal*, pp. 1–17, 2020, doi: 10.1016/j.ahj.2020.07.009.

[33] A. Dudchenko, M. Ganzinger, G. Kopanitsa, "Diagnoses Detection in Short Snippets of Narrative Medical Texts," *Procedia Computer Science*, vol. 156, pp. 150 – 157, 2019, doi: 10.1016/j.procs.2019.08.190.

[34] J. Prada, Y. Gala, A. Sierra, "Covid-19 mortality risk prediction using x-ray images," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 6, pp. 7–14, 2021, doi: 10.9781/ijimai.2021.04.001.

[35] K. Negi, A. Pavuri, L. Patel, C. Jain, "A novel method for drug-adverse event extraction using machine learning," *Informatics in Medicine Unlocked*, vol. 17, p. 100190, 2019, doi: 10.1016/j.imu.2019.100190.

[36] "Detection of unexpected findings in radiology reports: A comparative study of machine learning approaches," vol. 160, p. 113647, 2020, doi: 10.1016/j.eswa.2020.113647.

[37] T. F. d. Toledo, H. D. Lee, N. Spolaôr, C. S. R. Coy, F. C. Wu, "Web System Prototype based on speech recognition to construct medical reports in Brazilian Portuguese," *International Journal of Medical Informatics*, vol. 121, pp. 39 – 52, 2019, doi: 10.1016/j.ijmedinf.2018.10.010.

[38] L. F. Donnelly, R. Grzeszczuk, C. V. Guimaraes, W. Zhang, G. S. B. III, "Using a Natural Language Processing and Machine Learning Algorithm Program to Analyze InterRadiologist Report Style Variation and Compare Variation Between Radiologists When Using Highly Structured Versus More Free Text Reporting," *Current Problems in Diagnostic Radiology*, vol. 48, no. 6, pp. 524 – 530, 2019, doi: 10.1067/j.cpradiol.2018.09.005.

[39] H. Bay, T. Tuytelaars, L. Van Gool, "Surf: Speeded up robust features," in *Computer Vision – ECCV 2006*, Berlin, Heidelberg, Germany, 2006, pp. 404–417, Springer Berlin Heidelberg.

[40] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, USA, 2011, p. 2564–2571, IEEE Computer Society.

[41] P. F. Alcantarilla, A. Bartoli, A. J. Davison, "Kaze features," in *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*, ECCV'12, Berlin, Heidelberg, 2012, p. 214–227, Springer-Verlag.

[42] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K. Q. Weinberger Eds., Curran Associates, Inc., 2013, pp. 3111–3119.

[43] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota, June 2019, pp. 4171–4186, Association for Computational Linguistics.

[44] M. Kusner, Y. Sun, N. Kolkin, K. Weinberger, "From word embeddings to document distances," vol. 37 of *Proceedings of Machine Learning Research*, Lille, France, 07–09 Jul 2015, pp. 957–966, PMLR.

[45] W. Boag, E. Sergeeva, S. Kulshreshtha, P. Szolovits, A. Rumshisky, T. Naumann, "Cliner 2.0: Accessible and accurate clinical concept extraction," in *ML4H: Machine Learning for Health Workshop at Advances in Neural Information Processing Systems*, NIPS '17, 2017.

[46] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, pp. 1234–1240, 09 2019, doi: 10.1093/bioinformatics/btz682.

[47] I. Beltagy, K. Lo, A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, Nov. 2019, pp. 3615–3620, Association for Computational Linguistics.

[48] A. E. W. Johnson, T. J. Pollard, S. J. Berkowitz, N. R. Greenbaum, M. P. Lungren, C.-y. Deng, R. G. Mark, S. Horng, "Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports," *Scientific Data*, vol. 6, p. 317, Dec 2019, doi: 10.1038/s41597-019-0322-0.

[49] D. Demner-Fushman, S. Antani, M. Simpson, G. R. Thoma, "Design and development of a multimodal biomedical information retrieval system," *Journal of Computing Science and Engineering*, vol. 6, no. 2, pp. 168–177, 2012, doi: 10.5626/JCSE.2012.6.2.168.

[50] V. I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions and Reversals," *Soviet Physics Doklady*, vol. 10, p. 707, Feb. 1966.

Elvira Amador-Domínguez

Elvira Amador-Domínguez received the B.Sc. degree in Computer Science (2017) and the M.Sc. in Artificial Intelligence (2018) from the Universidad Politécnica de Madrid. Her B.Sc final thesis was awarded as one of the best thesis of the year 2017. She is currently a PhD Candidate at the Universidad Politécnica de Madrid, founded by a grant of the own university. Her prime fields of research include knowledge representation, deep learning, knowledge integration and explainability. She has also participated in the European Project AI4EU, as well as in a national educational innovation project.

Emilio Serrano

Emilio Serrano received the M.Sc. degree in computer science (2006) and the Ph.D. degree, with European mention and Extraordinary Ph.D. Award in artificial intelligence (2011), from the University of Murcia, Spain. He has also been a Visiting Researcher with The University of Edinburgh, the University of Oxford, and the National Institute of Informatics in Tokyo. He is currently an Associate Professor with the Department of Artificial Intelligence, Universidad Politécnica de Madrid (UPM). He is also Secretary of the Ph.D. in Artificial Intelligence at UPM. His main research line is the Social and Explainable Artificial Intelligence for Smart Cities. His scientific production includes more than 80 publications, highlighting over 25 articles in the JCR. He lectures deep learning and social network analysis among other courses; and, has been principal investigator in three educational innovation projects in data science. He has also participated in several European and National funding programs such as FP7 research projects (smartopendata, eurosentiment, and omelette) and H2020 research projects (slidewiki and AI4EU).

Daniel Manrique

Daniel Manrique received the B.S. and Ph.D. degrees in computer science from the Universidad Politécnica de Madrid (UPM), Madrid, Spain, in 1997 and 2001, respectively. He has been a visiting researcher with the University of Sunderland and Trinity College of Dublin. He is currently a member of the artificial intel ligence lab workgroup and an Associate Professor of computing with the Departmento de Inteligencia Artificial, UPM's School of Computing. His major fields of study and research are the subsymbolic artificial intelligence, its synergies with the symbolic domain, and diverse applications such as the medical area. He has published more than 70 research works on these topics in international journals, conferences, books, and book chapters. Dr. Manrique has participated as a researcher in several European, National, and regional research projects related. He is a member of the international program committee of several international congresses and acts as a reviewer of impact journals in the Journal Citation Report.

Javier Bajo

Dr. Javier Bajo, full professor at the Department of Artificial Intelligence, Computer Science School at Universidad Politécnica de Madrid (UPM), holds (since 03/05/2019) the position of Director of the UPM AI.nnovation Space Research Center in Artificial Intelligence. He was Director of the Department of Artificial Intelligence (20/05/2016-19/10/2017) at UPM, Secretary of the PhD in Artificial Intelligence at UPM (23/06/2016-19/10/2017) and Coordinator of the Research Master in Artificial Intelligence at UPM (18/02/2013 - 20/05/2016). He also holds the position of Director of the Data Center at the Pontifical University of Salamanca (13-10/2010 - 08-11-2012), with 21 employees. His main lines of research are Social Computing and Artificial and Hybrid Societies; Intelligent Agents and Multiagent Systems, Ambient Intelligence, Machine Learning. He has supervised 11 Ph.D thesis, participated in more than 50 research projects (in most of them as principal investigator) and published more than 300 articles in recognized journals (81 JCR papers) and conferences. His h-index is 39. He is founder of the PAAMS series of conferences and is an IEEE, ACM and ISIF member.