Expert Systems With Applications 64 (2016) 56–68

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Combination of multiple diagnosis systems in Self-Healing networks

David Palacios, Emil J. Khatib, Raquel Barco*

Communications Engineering Dept., University of Málaga, 29071, Málaga, Spain

ARTICLE INFO

Article history:
Received 8 January 2016
Revised 20 July 2016
Accepted 21 July 2016
Available online 22 July 2016

Keywords:
LTE
Self-healing
Root cause analysis
Self-organizing networks (SON)
Hybrid ensemble classifier
Automatic fault identification

ABSTRACT

The Self-Organizing Networks (SON) paradigm proposes a set of functions to automate network management in mobile communication networks. Within SON, the purpose of Self-Healing is to detect cells with service degradation, diagnose the fault cause that affects them, rapidly compensate the problem with the support of neighboring cells and repair the network by performing some recovery actions.

The diagnosis phase can be designed as a classifier. In this context, hybrid ensembles of classifiers enhance the diagnosis performance of expert systems of different kinds by combining their outputs. In this paper, a novel scheme of hybrid ensemble of classifiers is proposed as a two-step procedure: a modeling stage of the baseline classifiers and an application stage, when the combination of partial diagnoses is actually performed. The use of statistical models of the baseline classifiers allows an immediate ensemble diagnosis without running and querying them individually, thus resulting in a very low computational cost in the execution stage.

Results show that the performance of the proposed method compared to its standalone components is significantly better in terms of diagnosis error rate, using both simulated data and cases from a live LTE network. Furthermore, this method relies on concepts which are not linked to a particular mobile communication technology, allowing it to be applied either on well established cellular networks, like UMTS, or on recent and forthcoming technologies, like LTE-A and 5G.

1. Introduction

The growing demand for mobile services with ever-increasing bandwidth and the expanding number of users make necessary the deployment of new and more efficient mobile communication networks over the existing ones (GSM, UMTS), such as Long-Term Evolution (LTE). However, the complexity of this heterogeneous scenario, which comprises several Radio Access Technologies (RAT), requires challenging maintenance and complex operational tasks. Mobile operators need to offer new demanding services without increasing either operational expenditures (OPEX) or capital expenditures (CAPEX). In order to deal with that problem, the 3rd Generation Partnership Project (3GPP) has proposed Self-Organizing Networks (SON) (3GPP (d)) as networks that include mechanisms to automate network procedures in order to help mobile operators with their management work, providing significant cost reduction. This automation of network management will also be essential in near and future technologies, like LTE-Advanced and 5G (3GPP (b)).

SON comprises three groups of functions: Self-Configuration, Self-Optimization and Self-Healing. The aim of the latter is to autonomously solve the problems that a cell, with service degradation or outage, could present (3GPP (e); Barco, Lázaro, and Muñoz (2012)). This is done by means of four stages:

- *Fault Detection:* Responsible for finding cells with problems, i.e., cells experiencing service outage or just suffering an unacceptable service degradation.
- *Diagnosis of the fault cause:* In this step, the actions to be performed in order to recover the system from the degradation it is suffering are decided. This step can be divided into two substages: Fault Identification, this is, identifying the fault cause based on observable symptoms such as Key Performance Indicators (KPI) and alarms; and Action Identification, which corresponds to the decision of what tasks to perform to recover the system normal performance.
- *Fault recovery:* In this step, the proposed solutions are carried out.
- *Fault compensation:* Since diagnosing the fault and repairing it normally takes some time, compensation aims to diminish the impact of the fault by changing parameters in neighboring cells.

* Corresponding author. Fax: +34952132027.
E-mail addresses: dpc@ic.uma.es (D. Palacios), emil@uma.es (E.J. Khatib), rbarco@uma.es (R. Barco).

http://dx.doi.org/10.1016/j.eswa.2016.07.030

This paper is focused on the diagnosis task, in particular in the fault identification, also called root cause analysis. Once a problem has been detected in a cell, root cause analysis identifies the fault cause given the value of performance indicators, alarms, counters, mobile traces, etc. In the context of cellular networks, some diagnosis systems have been recently proposed. Barco, Díez, Wille, and Lázaro (2009) and Barco, Lázaro, Wille, Díez, and Patel (2009) proposed diagnosis systems based on Bayesian Networks. Szilágyi and Nováczki (2012) used a scoring system in order to determine how well a specific case fits a diagnosis. Nováczki (2013) enhanced the previous system by adding profiling techniques. The method in Khatib, Barco, Gómez-Andrades, and Serrano (2015) was based on fuzzy logic and genetic algorithms. Gómez-Andrades, Muñoz, Serrano, and Barco (2016) proposed a diagnosis system based on Self-Organized Maps (SOM).

Each of the previous methods has its pros and its cons. In practice, this makes the selection of the diagnosis technique cumbersome when the aim is to deploy a automatic diagnosis system in a real network. Furthermore, once the technique has been decided, e.g., fuzzy logic, operators normally design several standalone diagnosis models. This is due to the fact that, firstly, different troubleshooting experts will build different models and secondly, when models are learnt from historical cases, different training datasets will result in different models.

To cope with the limitations of standard classifying systems in terms of accuracy and dataset-dependent performance, ensembles of classifiers arose. Within these, homogeneous and heterogeneous (commonly known as hybrid) ensembles of classifiers may be found, where the former stand for the ensemble of classifiers of the same kind and the latter stand for the combination of different kinds of systems and datasets. Despite homogeneous ensembles have been widely studied and as of today still are extensively used in different fields (Begum, Chakraborty, & Sarkar, 2015; Liu, Chen, Song, & Han, 2009; Shen & Chou, 2006; Wiezbicki & Ribeiro, 2016). In this paper, a method for the generalized combination of multiple diagnosis systems based on a hybrid ensemble approach is proposed and tested in the context of cellular networks, which to the authors' knowledge is a research area still to be explored. The proposed work describes a method to gather, combine and use the knowledge held by any kind of expert system in any field that makes use of a classifying or diagnosis system. In this work, the proposed method is applied in the fault cause diagnosis in cellular networks, where the expertise may be provided either by a human troubleshooting expert or by a database of cases assessed by automatic diagnosis systems. The proposed method allows combining diagnosis systems in a wide sense, being able to merge both several diagnosis models (expertise) and the tools used for their application (automatic diagnosis techniques) in the form of supervised or unsupervised classifying systems.

Up to now, hybrid ensembles of classifiers are mainly based on a set of baseline systems which must first assess the cases under test and, consequently, provide partial diagnoses which are finally combined into a final decision using a majority vote scheme (Ciocarlie, Lindqvist, Nováczki, & Sanneck, 2013; Gandhi & Pandey, 2015; Wei et al., 2014). This procedure requires a relatively high number of diagnosis techniques to be run in the test stage and, therefore, a noticeable expenditure of computational and time resources. The proposed work, however, presents a method which allows combining the diagnoses that the standalone diagnosis systems would output for a case under test without actually needing them to be run, thus lightening the computational weight of the test stage.

The main contributions of this paper are:

- A method to combine any number and kind of different standalone classifiers as well as different sources of expert knowledge in order to get an enhanced performance compared to that of the base classifiers. In the context of troubleshooting in cellular networks this comprises the combination of several diagnosis models and techniques for the automatic diagnosis.
- A method to lighten the computational cost of the evaluation stage in hybrid ensembles of classifiers. This work proposes a scheme to model and emulate the behavior of every standalone classifier so these need not to be continuously queried before combining their partial diagnoses.

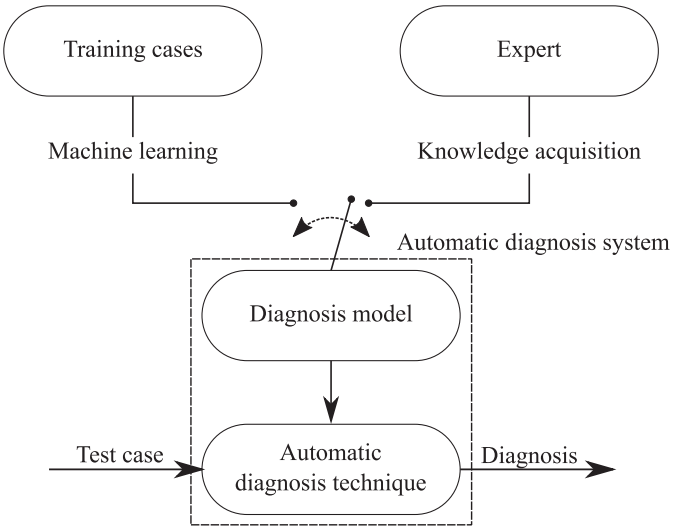


Fig. 1. Scheme of an automatic diagnosis system.

This paper is organized as follows. Section 2 presents the problem formulation. Section 3 introduces the proposed method for combining multiple baseline diagnosis systems. In Section 4 results are analyzed by means of both a network simulator and data from a live LTE network. In Section 5 the future lines of work are outlined. Finally, Section 6 summarizes the main conclusions.

2. Problem formulation

2.1. Root cause analysis in mobile communications networks

In the same way that a patient is diagnosed by a doctor based on the symptoms he shows, the status of a communications network may be diagnosed based on a set of performance indicators. This diagnosis task, also called root cause analysis or troubleshooting, is often carried out by human experts using their knowledge on the underlying relations that the observed indicators and the status of the network have. However, the number of symptoms (counters, alarms, KPIs, call traces, etc.) and possible fault causes the expert has to deal with increases as networks grow in size and complexity, which makes this task to become a very difficult and time consuming issue.

Furthermore, the current manual troubleshooting is a layered task, guided by a Trouble Ticket (TT) system. In this problem solving system, a group of specialists tries first to diagnose and solve the problem by performing some simple checks. If they can not find the root of the problem, this is raised to a more specialized team (and so on), which performs a deeper study on the symptoms the case exhibits and resorts to field engineers in case they need to make some on site checks.

As a response to this more and more inefficient procedure, automatic diagnosis systems arose in an attempt of imitating the way of acting of troubleshooters. Fig. 1 shows the basic scheme of a system for automatic diagnosis. It is composed of an au-

tomatic diagnosis technique and a diagnosis model. The first is an artificial intelligence system that outputs a diagnosis taking a set of symptoms, e.g., (KPIs) from a test case as its input. The second represents the knowledge a human expert would have on the underlying relations between the symptoms and the fault causes and may take different forms depending on the diagnosis technique it is destined to work with. For example, a diagnosis model may consist of the parameters (e.g., prior probabilities and probability density functions) required by a given diagnosis technique (e.g., bayesian classifier) or a set of rules for other techniques (e.g., Case Base Reasoning, CBR). As it can be seen in this figure, the diagnosis model may be built from a set of training cases by means of a machine learning algorithm or by troubleshooting experts by gathering their knowledge. The proposed method aims to combine the knowledge acquired by any number and kind of diagnosis models and automatic diagnosis techniques in an attempt to reduce the errors in fault detection and diagnosis.

2.2. Automated diagnosis from the classification theory

A diagnosis system is a method that given a set of indicators or symptoms (called *case* hereafter) intends to infer the cause that provoked them. In this sense, a diagnosis system acts as a classifying system in which the attributes from the cases to be classified correspond to the symptoms from the case to be diagnosed, and the classes to be assigned correspond to the causes to be inferred. This is an issue long time investigated in data mining theory (Wu et al., 2008), and many types of classifiers have been developed over the years in an attempt to get the maximum information the cases under diagnosis could provide. However, no algorithm has proven to be clearly better than the rest for all kinds of input data by now. One reason for the increasing efforts in the related research is that the performance of a classifier normally depends on the nature and distribution of the data it has to work with. For this reason, the present paper focuses not only on combining different diagnosis models but on offering the possibility to combine multiple classifiers in the form of automatic diagnosis techniques.

Let us assume we have a set of M fault causes to diagnose and R diagnosis systems (either diagnosis model or technique) to combine, and that each of these systems can have a subset of these causes as their output, namely, W^r for the system r. In this scenario, the set of causes a diagnosis system can identify may be different from one system to another. This can be seen in (1), where each row stands for a W^r and the element w_m^r stands for the mth fault cause, diagnosed by the rth system. According to this, each row may be different from another.

$$\begin{bmatrix} w_1^1 & \dots & w_m^1 & \cdots & w_M^1 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ w_1^r & \dots & w_m^r & \cdots & w_M^r \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ w_1^R & \dots & w_m^R & \cdots & w_M^R \end{bmatrix} \tag{1}$$

In a diagnosis system, a case, $\bar{x}$, is characterized by its symptoms, x_n, where $\bar{x} = \{x_1, x_2, \dots, x_N\}$, having a total of N possible symptoms. However, each diagnosis system may consider only a subset of these symptoms, namely, N^r for the diagnosis system r.

In the context of diagnosis systems for mobile communication networks a case corresponds to an observation or measurement from the network; a symptom may be an event counter, a Key Performance Indicator (KPI), a call trace or an alarm and the causes are seen as the network states, among which the normal and several fault states may be distinguished. In this paper, some results from theory of classifiers is used, extended and applied in this context in an attempt of combining the knowledge acquired by these R diagnosis systems, developing a more reliable and accurate root cause analysis system for communication networks.

2.3. State-of-the-art in ensemble-based classification algorithms

This section aims to provide a brief survey on the most recently proposed ensemble-based systems, most of which have been used in classifying tasks in areas not related to mobile communications.

Ensembles of classifiers may nowadays be classified into homogeneous and heterogeneous or hybrid. The first stand for those ensembles which put together instances of classifiers of the same kind, e.g., several k-Nearest Neighbor (kNN) classifiers. Conversely, in heterogeneous ensembles a set of classifiers of different kind are put together, e.g., a kNN and a NN (Neural Network). This is the scope of the present work, as the latter also allow the combination of different sources of expert knowledge within a single enhanced diagnosis system.

One of the earliest works on ensemble methods proposed to partition the feature space (i.e., the vector space in which the features of the cases to be diagnosed are defined) and to assign each part to a different classifier which is supposed to be the best for this subset of cases (Dasarathy & Sheela, 1979). This idea has been widely explored and has given birth to the so-called *mixture of experts* algorithm (Jacobs, Jordan, Nowlan, & Hinton, 1991; Yuksel, Wilson, & Gader, 2012), being the paradigm for the *classifier-selection* type of ensemble methods. Under this approach, only one classifier is working at the same time and its selection is determined by the partition the case under test belongs.

Conversely, in *classifier-fusion* methods all classifiers are usually trained over the entire feature space. The classifier combination process involves merging the individual classifiers to obtain a system that outperforms the standalone classifiers. This is the basis for the widely used bagging and boosting predictors (Breiman, 1996; Freund & Schapire, 1997), being AdaBoost an example of the latter and one of the most known and used algorithms for classifying nowadays. Classifier fusion methods can also be divided into those which work with classification labels only and those which make use of a continuous valued output for each classifier for every class. In this case, the outputs can be seen as the support an expert gives to a class in terms of the class-conditional posterior probabilities (Kuncheva, 2002).

Some examples of ensemble methods as enhanced systems for fault disclosure can be found in the literature with many different purposes. In Liu et al. (2009), an homogeneous ensemble of neural networks with cross-validation for fault diagnosis of analog circuits with tolerance is proposed. In Shen and Chou (2006), several kNN classifiers are put together on a majority-vote ensemble to classify the patterns that several proteins may exhibit when folded. In Begum et al. (2015), an homogeneous ensemble of SVM (Support Vector Machine) is proposed to identify different types of cancer from a genetic analysis. Wiezbicki and Ribeiro (2016) proposes an homogeneous ensemble of neural networks, combined by means of a weighted majority vote in a sensor network for the classification of gases.

Regarding the most recent works on hybrid ensembles of classifiers, in Wei et al. (2014) n ensembles are made up by combining $3n$ baseline classifiers. Each ensemble comprises three supervised methods: a decision tree, a support vector machine and a kNN algorithm. In each ensemble, the diagnoses from these baseline classifiers are fused applying a weighted majority vote, where each vote is weighted by the performance each individual classifier

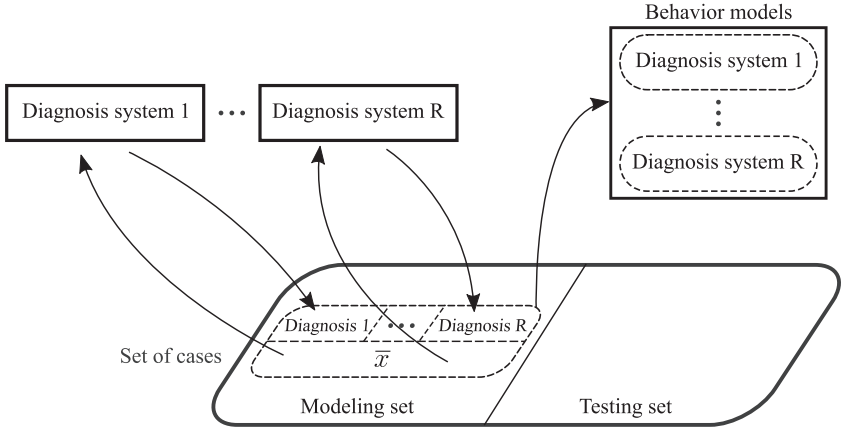


Fig. 2. Proposed method for combining diagnosis systems. Stage 1: Construction of the behavior models.

shows during a prior training stage. Then, the n resulting diagnoses are combined into a final diagnosis applying a non-weighted majority vote. In this case, all the baseline classifiers must be supervised diagnosis systems, as their performance must be previously known in order to weigh their votes in the first stage. Unlike this, the proposed method allows the user to combine any kind of diagnosis system, either supervised or unsupervised ones. And even more important, regarding the operation stage, in Wei et al. (2014), whenever a new case is to be diagnosed it must pass through two steps, one of them made up of $3n$ systems which must first each output a diagnosis, resulting in a high computational cost. The method in the proposed work, however, needs the test cases to be assessed only by one step, which, furthermore, only consist of some algebraic calculations. Once the training stage has been performed, new cases will be diagnosed at a minimum computational cost.

As for Gandhi and Pandey (2015), a two-step method is again proposed. The first step consists of a learning stage for the base classifiers and the second step consists of a majority vote-based combining stage. Again and similar to Wei et al. (2014), every baseline classifier is required to first diagnose every new case in the application step, which results in a high computational cost compared to that from the application (test) step in the proposed method.

In the context of cellular networks, Ciocarlie et al. (2013) proposes a hybrid ensemble of classifiers to detect anomalies in the performance indicators of a cell. This work is focused on the fault detection. Unlike this, the proposed work does not just find a performance degradation, but identifies the fault cause behind it. Regarding its implementation, this method relies on the use of a pool of models. New models are added to this pool whenever a change in the configuration parameters of the network takes place. A number of $N_{CM} \times \left(N_{univariate} \times N_{KPI}^{U} + N_{multivariate} \times N_{KPI}^{G}\right)$ models, and thus, instances of automatic techniques must be assessed for every single new case under test. In this expression, N_{CM} stands for the number of sets of network configuration parameters considered; N_{KPI}^{U} and $N_{univariate}$ stand for the number of univariate techniques considered and the number of KPIs acting as their input in each model; and $N_{multivariate}$ and N_{KPI}^{G} stand for the number of multivariate techniques used and the number of groups of KPIs considered in each model. Like in Ciocarlie et al. (2013), Wei et al. (2014) and Gandhi and Pandey (2015), before an ensemble decision can be made, a high number of baseline classifiers must be first queried. And again similarly to Ciocarlie et al. (2013), according to Wei et al. (2014) and Gandhi and Pandey (2015), all the partial decisions meet at a combining stage based in a weighted majority vote.

To the authors' knowledge, no ensemble method for fault cause diagnosis in cellular networks has been proposed as of today.

3. Method for combining multiple automatic diagnosis systems

In this section, a method for combining the knowledge acquired by any number and kind of standalone automatic diagnosis systems by means of a classifier-fusion scheme is proposed. The proposed method consists of two stages: the construction of the behavior models of the automatic diagnosis systems, Section 3.1, and the combination of these models in order to make a more accurate diagnosis on the cases from a testing set, Section 3.2. This can be seen in Figs. 2 and 5. Before this method can be applied, two sets of N-dimensional cases must be distinguished: the modeling set and the testing set, where each of these N dimensions stands for a working KPI. The modeling set will be used in the first stage and the testing set in the second.

3.1. Construction of the behavior models

The baseline diagnosis systems are to be combined by means of mixing their models of behavior, which need to be extracted first.

Once the diagnosis model from each diagnosis system has been built (either from training cases via a machine learning method (Khatib, Barco, Gómez-Andrades, Muñoz, & Serrano, 2015), or from the experts' knowledge (Gómez-Andrades et al., 2016) each diagnosis system can start the classification (Fig. 1). In this stage, every case from the modeling set is diagnosed by the R systems. That is, each system assigns to each case one of the M possible fault causes; in particular, one of the causes that system can discern. This can be seen in Fig. 2, where the case $\bar{x}$ acts as the input for the R systems and, in turn, they assign it R diagnosis labels. If the system r diagnoses the case $\bar{x}$ with the cause m, this case receives the label w_m^{r*}. In this way, each diagnosis system makes a different partition of the modeling set into $|W^{r*}|$ disjoint subsets, whose maximum is $|W^r|$, that is, the number of causes that system considers (Fig. 3), where $|A|$ is the number of elements in the set A. This leads to finally identify M^* different causes, being M^* the union of W^{r*} over r, with $M^* \leq M$. According to this, a new matrix from (1) may be written, substituting every row (i.e., every W^r) by its corresponding W^{r*}. Each row would represent one of the partitions of the modeling set and each column would represent how a cause "is seen" by each diagnosis system regarding the KPIs the cases belonging to that w_m^r exhibit.

It should be noticed that each of these M^* subsets contains a number of $|N^r|$-dimensional cases. At this point, the behavior of the diagnosis system r is modeled through the estimation of the

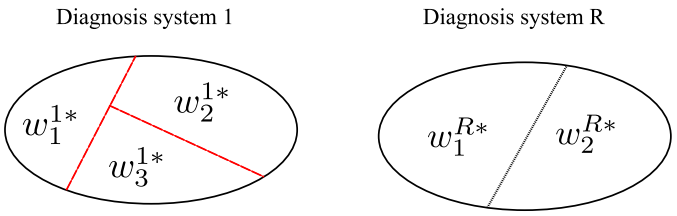


Fig. 3. Modeling set divided into different subsets by means of two different partitions: on the left, the partition the first diagnosis system makes, having $W^1 = \{w_1^1, w_2^1, w_3^1\}$ with $|W^1| = |W^{1*}| = 3$; on the right, the partition the diagnosis system R makes, having $W^R = \{w_1^R, w_2^R, w_3^R\}$ and $W^{R*} = \{w_1^{R*}, w_2^{R*}\}$. In this last case, the diagnosis system R only diagnosed the causes 1 and 2 although being able of also identifying the fault cause 3.

Table 1
Families of PDFs considered for the estimation of $p(x_n|w_m^{r*})$.

Distribution	PDF	Parameters
Beta	$\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1}(1-x)^{b-1}$	a, b
Normal	$\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$	μ, σ
Log-normal	$\frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\ln(x-\mu)}{\sigma}\right)^2\right)$	μ, σ
Exponential	$\lambda \exp(-\lambda x)$	λ
Gen. extreme value	$\frac{1}{\sigma} t(x)^{\xi+1} \exp(-t(x))$, $t(x) = \begin{cases} \left(1+\left(\frac{x-\mu}{\sigma}\right)\xi\right)^{-\frac{1}{\xi}} & \xi \neq 0 \\ \exp(-(x-\mu)/\sigma) & \xi = 0 \end{cases}$	μ, σ, ξ
T-location	$\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)\sqrt{\pi\nu}\sigma}\left(1+\frac{1}{\nu}\left(\frac{x-\mu}{\sigma}\right)^2\right)^{-\frac{\nu+1}{2}}$	ν, μ, σ
Nakagami	$\frac{2m^m}{\Gamma(m)\Omega^m} x^{2m-1} \exp\left(-\frac{m}{\Omega}x^2\right)$	m, Ω
Gamma	$\frac{1}{\Gamma(k)\theta^k} x^{k-1} \exp\left(-\frac{x}{\theta}\right)$	k, θ
Logistic	$\frac{\exp\left(\frac{x-\mu}{s}\right)}{s\left(1+\exp\left(-\frac{x-\mu}{s}\right)\right)^2}$	μ, s
Log-logistic	$\frac{(\beta/\alpha)(x/\alpha)^{\beta-1}}{\left(1+(x/\alpha^\beta)\right)^2}$	α, β
Weibull	$\frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} \exp\left(-\frac{x}{\lambda}\right)^k$	λ, k
Rayleigh	$\frac{x}{\sigma^2} \exp\left(-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right)$	σ
Rice	$\frac{x}{\sigma^2} \exp\left(-\frac{x^2+\nu^2}{2\sigma^2}\right) I_0\left(\frac{x\nu}{\sigma^2}\right)$	ν, σ

statistical distributions of the N^r KPIs for the cases belonging to W^{r*}. That is, the behavior of each diagnosis system is modeled by means of $N^r \times M^*$ PDFs. The estimated statistical distribution of the nth KPI for the subset of cases diagnosed as m by the diagnosis system r is $p(x_n|w_m^{r*})$. The choice of the PDF that estimates each one of these distributions is done according to the maximum likelihood (ML) criterion. To do so, some families of PDFs are considered in the fitting procedure (Table 1). In a first step, the distribution of the KPI x_n from the cases labeled as w_m^{r*} is fitted attending to the ML criterion with each one of the considered families of PDFs. This results in a set of candidates for estimating its distribution. These PDFs are then sorted by their likelihood and the one with the maximum value is chosen to be the estimation for the KPI.

The reason for considering these families of PDFs is to get the better estimation of the distribution of the KPI x_n given its belonging to w_m^{r*}. Fig. 4a shows a normalized histogram of the KPI "95th percentile RSRP" from the cases labeled as w_m^{r*}. In this figure, two families of PDFs have been used in an attempt of fitting the underlying histogram, the normal and the generalized extreme value. As it can be seen, the latter fits it better, resulting in a higher value in a likelihood-ratio test.

While some KPIs are counters and they do not have an upper limit, there are others that are inherently bounded, as they are defined as a ratio. Normally, the beta PDF is used to fit these KPIs, usually limited between zero and one (Barco, Lazaro, Diez, & Wille, 2008). KPIs like the retainability or the accessibility often reach these extreme values making the resulting fitted beta present asymptotes in these values. To avoid this issue the used beta function β' is slightly different from that from Table 1, β. In this case,

$$\beta'(x) = (1 - P_0 - P_1)\beta(x) + P_0/h_\beta \delta(x) + P_1/h_\beta \delta(x-1), \tag{2}$$

where $\beta(x)$ stands for the distribution fitted to a set with no extreme values; P_0 and P_1 stand for the relative frequency of cases with value 0 and 1 respectively; δ stands for the Dirac's delta and h_β stands for the step (the resolution) when computing β'. This can be seen in Fig. 4b, where a normalized histogram for the KPI retainability is shown.

3.2. Combination of behavior models

This stage uses the cases from the testing set. In the previous stage, the estimated functions have been seen as conditional probability density functions, that is, functions that express how the KPIs are distributed over the cases diagnosed with a given cause by a given system. However, this set of functions may be seen as likelihood functions by just changing the approach. From this point of view, the function depends on w_m^{r*} given that an observation of the random variable x_n (that is, the nth KPI) has taken place.

Now, assuming the KPIs are independent among each other, a joint probability function of w_m^{r*}, that is, $p(\bar{x}|w_m^{r*})$, may be written as

$$p(\bar{x}|w_m^{r*}) = \prod_{n \in N^r} p(x_n|w_m^{r*}). \tag{3}$$

Given (3), and assuming that the prior probability of each cause, $P(w_m^{r*})$ is given by $\frac{|w_m^{r*}|}{|W^{r*}|}$, the *a posteriori* probability for a diagnosis system r to diagnose a case with the cause m given its KPIs are $\bar{x}$ (i.e., $P(w_m^r|\bar{x})$) can be calculated by just applying the Bayes' theorem. That is,

$$P(w_m^r|\bar{x}) = \begin{cases} \dfrac{p(\bar{x}|w_m^{r*})P(w_m^{r*})}{\sum_{w_i^{r*} \in W^{r*}} p(\bar{x}|w_i^{r*})P(w_i^{r*})} & if \quad P(w_m^{r*}) > 0 \\ 0 & if \quad P(w_m^{r*}) = 0 \end{cases} \tag{4}$$

At this point, some diagnosis system may have not diagnosed a given cause as seen in Fig. 3. In such case, $P(w_m^{r*})$ and thus (4) would result equal to zero. In any case, $M \times R$ *a posteriori* probabilities may be distinguished. Fig. 5 shows this when a case $\bar{y}$ from the testing subset is to be diagnosed. As it can be seen in this figure, the KPIs from the case $\bar{y}$ act as input values in the behavior models of the R diagnosis systems, i.e., the probability functions $p(\bar{y}|w_m^{r*})$ for $w_m^{r*} \in W^{r*}$ and $r = 1, \ldots, R$. Then, the *a posteriori* probabilities $P(w_m^r|\bar{y})$ are computed using these together with $P(w_m^{r*})$ by means of the Bayes' theorem.

Now, these $M \times R$ *a posteriori* probabilities together with the prior probabilities can be combined over R using some algebraic functions, producing M probabilities of the kind $P(w_m|\bar{y})_{Rule_t}$ per function used, where m again stands for the cause and t is an index for the rule used in the combination, that is,

$$P(w_m|\bar{y})_{Rule_t} = f_{Rule_t}\left(P(w_m^1|\bar{y}), \ldots, P(w_m^R|\bar{y}); P(w_m)\right). \tag{5}$$

where $P(w_m)$ is defined as the average of $P(w_m^{r*})$ over r.

Some rules for the combination of *a posteriori* probabilities given by several classifying systems are proposed in Kittler, Hatef, Duin, and Matas (1998) and studied further in Kuncheva (2002). In the first, those rules are derived from a maximum *a posteriori* (MAP) estimation in a multiple random variable scenario in an attempt of lightening the efforts of computing several joint probability density functions. These rules are summarized in Table 2.

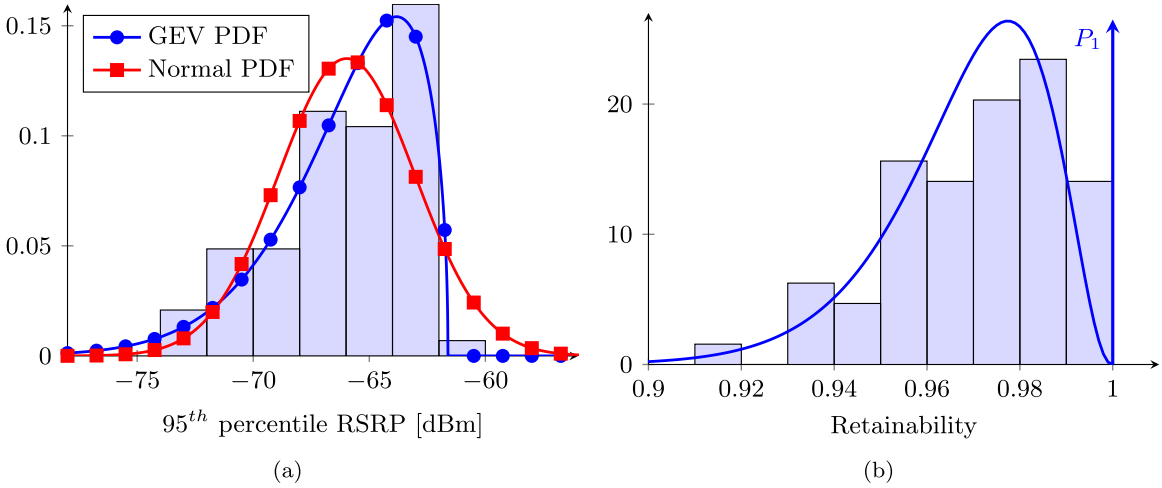


Fig. 4. (a) Normalized histogram for the KPI 95th percentile RSRP and two fitted PDFs: a generalized extreme value PDF in blue (round markers) and a normal PDF in red (square markers). (b), Normalized histogram for the KPI Retainability and a β' PDF estimation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

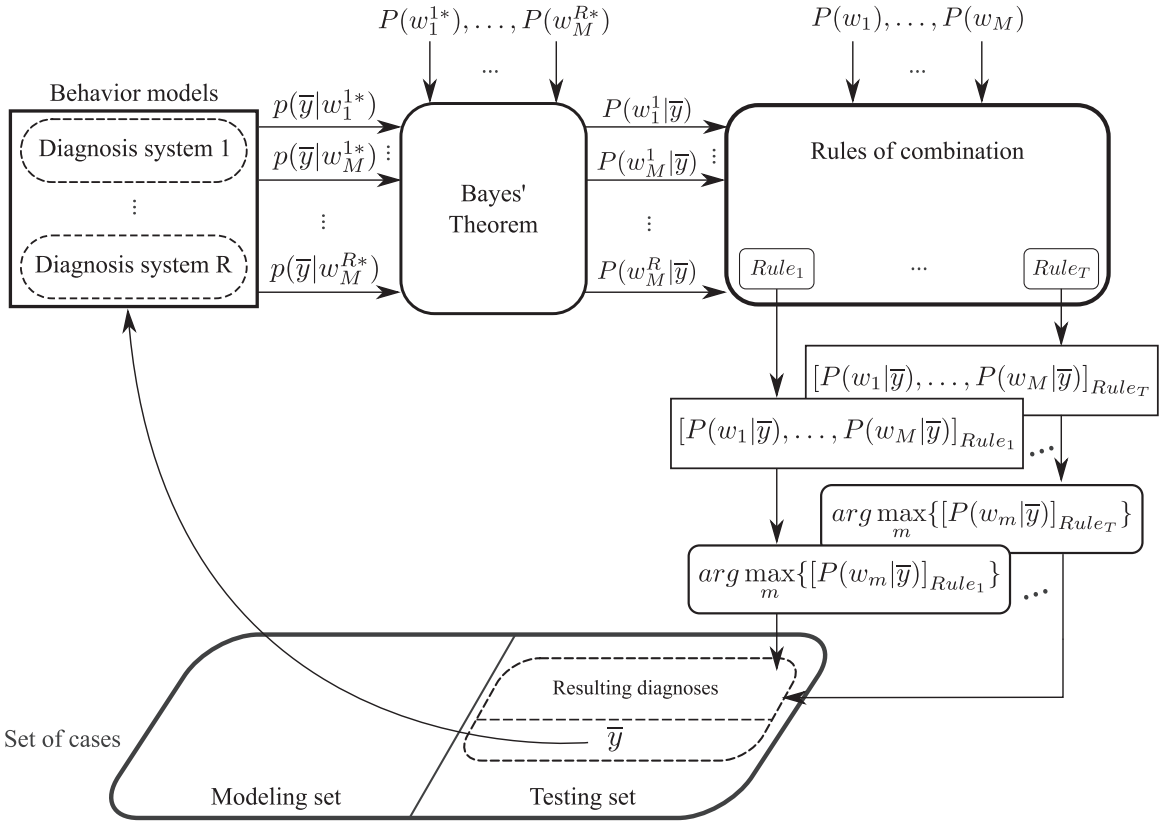


Fig. 5. Proposed method for combining diagnosis systems. Stage 2: Combining the behavior models.

As this point, the fault cause with the maximum *a posteriori* probability is taken as the final diagnosis per each rule of combination, d_t. That is,

$$d_{Rule_t} = \arg\max_m\{P(w_m|\overline{y})_{Rule_t}\}. \quad (6)$$

Note that a situation with $M^* < M$ means that there is at least one fault cause that have not been identified by any system. In this case, it would be impossible for it to be finally diagnosed in consequence.

4. Proof of concept

In this section, the proposed method is assessed by combining two different diagnosis models. In the first test, each model is provided by a different expert; in the second test, each model comes from using different machine learning algorithms for building the diagnosis models, provided the same set of training cases.

The proposed method has been evaluated and compared to the baseline systems by means of the following figures of merit:

Table 2
Algebraic rules for the combination of a posteriori probabilities.

Rule	$P(w_m\|\overline{y})$
Product rule	$P(w_m)^{-(R-1)}\prod_{r=1}^{R}P(w_m^r\|\overline{y})$
Sum rule	$(1-R)P(w_m)+\sum_{r=1}^{R}P(w_m^r\|\overline{y})$
Max rule	$(1-R)P(w_m)+R\max_{r=1}^{R}\{P(w_m^r\|\overline{y})\}$
Min rule	$P(w_m)^{-(R-1)}\min_{r=1}^{R}\{P(w_m^r\|\overline{y})\}$
Median rule	$\underset{r=1}{\overset{R}{med}}\{P(w_m^r\|\overline{y})\}$

- Diagnosis Error Rate (DER): it is the ratio of problematic cases diagnosed as a fault cause different to the real one (misclassified cases), N_{MPC}, to the total number of problematic cases, N_{PC}.
- False Positive Rate (FPR): it is the number of normal cases diagnosed as problematic cases, (N_{FP}), to the total number of normal cases, (N_{NC}).
- False Negative Rate (FNR): it is the number of problematic cases diagnosed as normal cases, N_{FN}, to the total number of problematic cases, N_{PC}. This is the most critical metric, as it gives an idea on how often the diagnosis system interprets there is no problem when actually some cells are suffering from malfunctioning.

Given these definitions, an Overall Error Rate (OER) may be defined as

$$OER = P_N \cdot FPR + P_{PR} \cdot (FNR + DER) \tag{7}$$

where P_N stands for the relative frequency of the normal cases and P_{PC} stands for the relative frequency of the faulty cases. This metric is useful to assess every method at a single glance. Since these figures of merit require the true cause to be known, the used testing set will include the real diagnosis.

4.1. Combination of diagnosis models devised by multiple experts

4.1.1. Scenario

In this test, cases are provided by an LTE RAN simulator (Muñoz et al., 2011). This simulator considers an LTE network composed of 57 macro-cells evenly distributed in space and grouped into 19 three-sector-sites. To perform this test, similar network configuration parameters to those used in Gómez-Andrades et al. (2015) and Gómez-Andrades et al. (2016) have been used. They can be seen in Table 3.

With this simulator, 1196 cases have been obtained. In this case, training cases are not needed since the diagnosis models have been defined by experts. It is assumed that a detection system is placed before the input of the diagnosis system, so that only the faulty cases are put under test, putting aside the cases belonging to a normal cause of functioning. Therefore, in this test only the DER is taken into account.

In this scenario, six typical RAN fault causes have been considered ($M = 6$):

- *Excessive downtilt:* This situation takes place when the coverage area for a cell is too small, making the signal level in the edge of the cell to be too weak and causing a high number of handover failures. The quality of the signal in the surroundings of the cell is also decreased.
- *Coverage hole:* A cell has a coverage hole in some point inside its area when the power received by the user at this point from any cell is not enough to hold the service. This excessive attenuation can be caused by either obstacles or a bad RF planning and it mainly produces a high number of call drops.
- *Inter-system interference:* This fault cause may occur due to other cellular networks, like WCDMA. It is not always an easy issue to solve, since the fault usually comes from an outer system. This fault normally causes both the SINR and the average throughput decrease.
- *Too late handover:* A too late handover takes place if a radio link failure occurs while the UE (User Equipment) is moving from one cell to another and the corresponding handover between these cells has not taken place yet. In that case, the UE will request the second cell a connection re-establishment using the physical cell ID of the first cell and its Common Radio Network Temporary Identifier (C-RNTI) in that first cell, which will alert the second cell a too late handover has occurred.
- *Excessive uptilt:* A cell suffers from excessive uptilt when its coverage area is larger than necessary, normally because of a bad configuration of the radiation parameters of the antennas. This situation can result in the overlapping of coverage areas from

Table 3
Simulation parameters for cells normal functioning.

Parameter	Configuration
Cellular layout	Hexagonal grid, 57 cells, cell radius 0.5 km
Transmission direction	Downlink
Carrier frequency	2.0 GHz
System bandwidth	1.4 MHz, 6 PRB (Physical Resource Block)
Frequency reuse	1
Propagation model	Okumura-Hata with wrap-around, Log-normal slow fading, σ_{sf} = 8 dB and correlation distance = 50 m
Channel model	Multipath fading, ETU model
Mobility model	Random direction, 3 kph
Service model	Full Buffer, Poisson traffic arrival
Base station model	Tri-sectorized antenna, SISO, $P_{TX_{max}}$ = 43 dBm, Downtilt = 9 °, Azimuth beamwidth = 70 °, Elevation beamwidth = 10 °
Scheduler	Time domain: Round-Robin, Frequency domain: Best Channel
Power control	Equal transmit power per PRB
Link Adaptation	Fast, CQI (Channel Quality Indicator) based, perfect estimation
Handover	Triggering event = A3, HOM (Handover Margin) = 3 dB, Measurement type = RSRP
Radio Link Failure	SINR < −6.9 dB for 500 ms, Mehlführer, Wrulich, Colom Ikuno, Bosanska, and Rupp (2009)
Traffic distribution	Evenly distributed in space
Time resolution	100 TTI (Transmission Time Interval) (100 ms)
Epoch & KPI time	100 s

Table 4
Parameters used for modeling fault causes in Section 4.1 and a priori probabilities for each cause.

Fault cause	Configuration	$P(\omega_m)$
Excessive downtilt	Downtilt = [16, 15, 14] °	0.18
Coverage hole	Δ_{hole} = [49, 50, 52, 53] dBm	0.09
Inter-system interf.	$P_{TX_{max}}$ = 33 dBm Downtilt = 15 ° Azimuth beamwidth = [30, 60] ° Elevation beamwidth = 10 °	0.1
Too late HO	HOM = [6, 7, 8] dBm	0.23
Excessive uptilt	Downtilt = [0, 1] °	0.21
Lack of coverage	$P_{TX_{max}}$ = [7, 8, 9, 10] dBm	0.19

possibly non-adjacent cells, producing a high number of handovers and call drops in this cell and its neighbors
- *Lack of coverage:* A user suffers from weak coverage when the Signal-to-Interference-Plus-Noise Ratio (SINR) measured in the cell is below the minimum level needed to maintain a planned performance requirement because the received power is low.

The simulation parameters used to model these degradations are shown in Table 4, as well as the *a priori* probability of these causes to take place, given by the experts. In this case, $P(w_m^{1*}) = P(w_m^{2*}) \neq 0 \quad \forall m$, so $P(w_m) = P(w_m^{1*}) = P(w_m^{2*})$. As it can be seen, several values have been used for modeling a single fault cause, according to lighter and more severe degradation.

In this test, seven observable features or KPIs ($N = 7$) have been used to discern among this set of causes:

- *Retainability*, given as a percentage. This performance indicator quantifies the ability of the cell to hold the service once accepted by the admission control. It gives an idea on how often a user experiences a call drop.
- *Handover success rate (HOSR)*, given as a percentage. This KPI measures the ability of the network to provide mobility to a user without losing its connection. It can be calculated as the ratio between the number of successful handovers and the total number of HO.
- *95th percentile RSRP*, given in dBm. The Reference Signal Received Power (RSRP) is defined as the linear average over the power contributions (in [W]) of the resource elements that carry cell-specific reference signals within the considered measurement frequency bandwidth.
- *5th percentile RSRQ,* given in dB. The Reference Signal Received Quality (RSRQ) is a signal quality indicator and is defined as the ratio

$$RSRQ = \frac{N_{PRB} \cdot RSRP}{RSSI}, \tag{8}$$

where N_{PRB} is the number of resource blocks of the E-UTRA carrier RSSI measurement bandwidth and RSSI stands for the total received power within the measurement bandwidth. This is, considering the power from the serving cell, the power of the co-channel serving and non-serving cells, the adjacent channel interference and any possible source of noise. In this paper, RSRQ is expressed in dB.
- *95th percentile SINR*, given in dB. The Signal-to-Interference-plus-Noise Ratio (SINR) is defined as the ratio between the power of the desired data signal and the sum of the powers of all inter-cell interferences and the noise. It is expressed in dB.
- *95th percentile distance*, given in km. This KPI measures the distance between users and their serving cell, expressed in km. It can be estimated attending to the transmission delay between them and gives an idea of the cell coverage area.
- *Average throughput*, given in kbps. In LTE systems, the user throughput depends on the SINR experienced by the user through the following equation, 3GPP (c),

$$T_k = (1 - BLER(SINR_k)) \cdot \frac{D_k}{TTI}, \tag{9}$$

where BLER is the Block Error Rate obtained from the users' SINR, D_k is the data block payload in bits of user k and TTI is the transmission time interval.

In order to show the impact a proper modeling may have in the diagnosis performance of the proposed method the proportion of cases used for the modeling to the testing set has been varied from 25% to 75%. To obtain more reliable results when the number of cases are scarce either in the testing or in the modeling set, 50 repetitions have been made per modeling-to-testing ratio, randomizing the cases assigned to each set. Then, the resulting diagnosis error rates have been averaged over the 50 repetitions.

Table 5
Diagnosis models for the diagnosis systems used in test 1: used thresholds.

KPI	Thresholds
Retainability	[0.973, 0.996]
HOSR	[0.899, 0.989]
RSRP [dBm]	[−76.9, −72.4]
RSRQ [dB]	[−18.8, −18.2]
SINR [dB]	[13, 14.5]
Throughput [kbps]	[96.2, 111.67]
Distance [km]	[0.838, 0.88]

4.1.2. The standalone classifiers

In this test, for a given technique of automatic diagnosis, two diagnosis models are combined, $R = 2$, where each of them is provided by a different expert. This test represents the usual case in cellular networks where each troubleshooting expert defines his own set of rules and KPI thresholds to identify problems. When deploying the diagnosis system in a network, according to the proposed method, instead of choosing one single model, the knowledge from both experts is fused by combining two diagnosis models. Furthermore, both diagnosis models comprise the six fault causes and the seven different KPIs described above. That is, $W^1 = W^2$ with $|W^1| = M$ and $N^1 = N^2$.

The artificial intelligence technique used for these tests is based on a Fuzzy Logic Controller (FLC) (Khatib, Barco, Gómez-Andrades, & Serrano, 2015). This system contains rules, which are composed of the antecedent (the "if ..." part) and the consequent (the "then ..." part), being the last the cause the fuzzy logic controller assigns to a case if the antecedent is fulfilled attending to the fuzzyfied observable features of the case. On the one hand, Table 5 shows the thresholds used in both diagnosis models. The lower limit stands for the value below which a KPI is considered to be low; the upper limits stands for the value above which a KPI is considered to be high. On the other hand, Table 6 shows the *if ..., then ...* rules that make up each diagnosis model, given by each expert. From left to right, each column below "KPI" in Table 6 corresponds to the KPIs shown in Table 5. H stands for a high value in that KPI and L for a low value. Regarding the numbering of the diagnoses, 1 means excessive downtilt; 2: coverage hole; 3: inter-system interference; 4: too late handover; 5: excessive uptilt and 6: lack of coverage.

4.1.3. Results

Table 7 shows the diagnosis error rates computed when the *Max rule* is used for combining (Table 2). In Table 7, the average diagnosis error rate and the rate of improvement are shown. This last rate represents the amount of repetitions (among the 50 that have been performed) in which the diagnosis error rate from

Table 6
Diagnosis models for the diagnosis systems used in test 1: used rules.

Diagnosis model 1								Diagnosis model 2							
KPI							Diag.	KPI							Diag.
L	L	H	L	–	H	L	1	–	–	H	L	–	H	L	1
H	H	–	L	L	H	L	1	L	–	H	H	–	L	H	2
L	–	–	H	H	–	H	2	L	–	H	H	H	–	H	2
L	–	–	H	L	L	H	3	L	–	–	H	L	L	H	3
L	L	H	–	L	L	H	3	L	–	H	–	L	L	H	3
–	–	H	H	H	H	–	4	L	–	H	H	L	L	–	3
–	H	H	–	H	H	–	4	L	H	–	H	L	L	–	3
H	–	H	–	H	H	–	4	–	H	H	H	L	L	H	3
–	–	H	–	H	H	H	4	L	L	–	–	–	H	H	4
H	H	–	–	H	–	H	4	L	L	–	L	–	–	H	4
H	H	–	L	–	–	H	4	L	L	H	–	H	L	–	4
H	H	–	–	–	H	H	4	L	L	H	–	H	–	H	4
H	H	H	–	–	–	H	4	L	L	L	L	L	L	–	4
–	–	H	L	H	–	H	4	–	–	L	H	–	L	H	5
–	–	H	L	–	L	H	4	H	H	–	H	–	L	H	5
L	L	H	L	–	–	H	4	H	H	L	–	L	L	H	5
–	–	L	–	L	L	H	5	–	–	–	L	L	H	L	6
–	–	L	–	L	H	L	6								
L	–	–	L	L	H	L	6								
–	H	–	L	L	H	L	6								
H	H	–	–	L	H	L	6								
L	L	L	L	L	–	L	6								

Table 7
Results of test 1: Combining two versions of the same classifying algorithm.

	Modeling-to-testing ratio		
	25%	50%	75%
Diagnosis syst. 1, average DER	13.81%	13.7%	13.65%
Diagnosis syst. 2, average DER	16.34%	16.13%	16.3%
Ens. Method: Max rule average DER	8.29%	5.92%	5.34%
Rate of improvement	60%	98%	100%

the ensemble method is lower than the best one provided by the baseline diagnosis systems. With a 25% of modeling-to-testing ratio only 60% of the iterations shows a better ensemble diagnosis error rate than the ones from its base diagnosis systems, showing, therefore, little improvement in the average diagnosis error rate. This result highlights how the scarcity of cases for modeling impacts on the classifying performance of the ensemble. However, if the number of cases used for modeling is doubled 98% of the iterations shows a better diagnosis error rate, which results also in a lower average diagnosis error rate. In case the modeling-to-testing ratio is set to 75% every diagnosis error rate provided by the ensemble method is lower than the lowest provided by its components, reaching a 5.34% on average. This means a DER of approximately 1/3 the lowest DER achieved by the standalone classifiers.

Regarding the DER of the standalone diagnosis systems, it can be seen how these are held over the modeling-to-testing ratio. This is because of the randomizing process executed over the labeled cases to be divided into the modeling and testing subsets. When this random permutation is performed a number of times and some subsets (two, in this case) are chosen blindly from this set, the averages of the amount of cases labeled with a given cause in each of these subsets tend to the ratios of the labels from the original set. This is a consequence of the law of large numbers. For this reason, the resulting averaged DER of these baseline systems is independent on the size of the subsets made from the original set of cases.

4.2. Combination of different diagnosis systems on a live network

Once the proposed method has been tested with cases provided by a simulator, a second test with cases from a real live LTE network has been performed. In this test, the diagnosis models built from two different machine learning algorithms have been combined.

Table 8
Main parameters of the real LTE network used in test two.

Parameter	Configuration
Network Layout	Urban area
Number of cells	8679
System bandwidth	10 MHz
Number of PRBs	50
Frequency reuse factor	1
Max. Transmitted Power	46 dBm
Max. Transmitted Power of UE	23 dBm
Horizontal HPBW (Half-Power Beam Width)	65°
HOM	3 dB
KPI Time Period	Hourly
Number of observed cells	45
Number of days under observation	6 days per cell (on average)
Size of the dataset	14,692 labeled cases

4.2.1. Scenario

An LTE network composed of more than 8000 different cells providing coverage to almost 4 million people has been analyzed. Its vastness makes many different cells to coexist and also a wide variety of problematic causes to come up. Table 8 summarizes the main parameters of the network. Among all the available candidates, 45 random cells have been chosen to represent the network behavior. These cells have been monitored for almost 6 days on average and their KPIs have been stored in an hourly basis. Taking into account that the state of a single cell varies substantially throughout the day due to the traffic fluctuation, several cases have been stored from each cell at different hours, resulting in a total of 14,692 cases. Once these cases were gathered, they were all labeled by the experts, distinguishing four groups of cases ($M = 4$): three kinds of problematic patterns and the normal cell functioning. The causes of malfunctioning that were found are:

- *Overload*: This fault cause is mainly distinguished by a high number of RRC connections in the cell, which makes the CPU processing load and the number of HO attempts raise conse-

Table 9
Prior probability of occurrence for the causes considered in test two, $P(w_m)$.

P(Overload)	P(Lack of cov.)	P(Non-operating)	P(Normal)
0.01	0.22	0.47	0.3

quently. The accessibility and retainability KPIs also hold values quite below the ones for a cell with normal functioning.
- *Lack of coverage*: This issue can be identified based on the number of bad coverage evaluation reports, which should be noticeably high.
- *Non-operating cell*: In this case, and only if the cell is reporting any KPI measurement, most of the reported measurements should be near zero: the retainability, the accessibility, the number of performed HO, the number of RRC connections or the number of coverage reports.

The *a priori* probability of occurrence of each class has been computed as the average of $P(w_m^{r*})$ over r within this selection, Table 9. From this table it should be noted that there are more faulty cases than healthy ones. This is because a previous non-perfect faulty cases detecting stage has been applied, which bypassed some normal cases that now are to be diagnosed as such.

At this point, a 20% of the total number of cases (holding the proportion shown in Table 9 between them) were used as a training set for the machine learning algorithms and the rest were used to conform the modeling and testing sets in a ratio that, as in Section 4.1.3, was varied along the test.

In this test six of the most representative KPIs in an LTE network have been chosen to discern between the possible diagnoses, $N = 6$:

- *Retainability*: described in Section 4.1.1.
- *Accessibility*: It is used to show the percentage of connections that have got access to that cell over the KPI time period. A low value in this KPIs means that many connections have been blocked during the access procedure.
- *Number of RRC connections*: It is the number of successfully established RRC connections. Related to the *Accessibility* KPI, it gives an idea of the amount of users served by the cell.
- *Number of Ping-Pong Handovers*: This KPI counts the number of ping-pong HO that takes place in the cell over the measurement time period. A high value in this KPI may mean a bad configuration in the handover policy, as the number of connections that goes back and forth over a cell and its neighbors is high for a single call.
- *Number of bad coverage reports*: It counts the number of times a cell is notified that the UE measured a signal level in which the requirements for the Event A2 takes place, 3GPP (a). This is, the measured signal level is under a certain threshold.
- *CPU average load*: It is the average CPU load due to the processes carried out by the cell over the KPI time period.

4.2.2. The standalone classifiers

In this test, the two used standalone classifiers share a similar diagnosis system, a fuzzy-logic controller, which diagnoses the cases attending to *if* ... , *then* ... rules. The difference resides in the algorithms they use for learning the rules they apply during the diagnosis process. The first is a genetic algorithm and the second is a *data driven* algorithm (Khatib, Barco, Gómez-Andrades, Muñoz, & Serrano, 2015; Khatib, Barco, Gómez-Andrades, & Serrano, 2015) respectively. In genetic algorithms, three main processes may be distinguished: reproduction, by means of which new individuals are created by either mutation or combination of the previously existing; evaluation, or the calculation of the probability of each individual to survive and reproduce, and selection, a process in which some individuals are chosen to survive and reproduce based on the results from the evaluation stage. Likewise, data driven algorithms first take a case from the training set and derives the fuzzy rule that covers it. Then, it looks for the cases covered by this rule and scores the rule attending to the number of covered cases. New incoming cases are taken until the training set is completely explored. Provided this set of scored rules, the algorithm then fuses them into a lower number of rules in a attempt of maximizing the number of cases (and therefore, the score) covered by the resulting fused rules. In these tests, it is assumed that not only faulty cases, but also some normal cases are inputs for the diagnosis stage. This can happen when there is no detection system before the diagnosis system or in the realistic situation in which the detection system has a given probability of error. As in Section 4.1.2, both systems take as possible output all the presented diagnoses making use of the six KPIs shown above. Table 10 shows the thresholds used for these KPIs to consider them high or low and Table 11 shows the rules each machine learning algorithm has derived from the testing set. As in Table 6, H stands for a high value of the KPI and L, for a low value. The KPIs are sorted in the same way as in Table 10 and the numbering of the diagnoses are 1: CPU overload; 2: lack of coverage; 3: non-operating cell and 4: normal functioning.

Table 10
Diagnosis models for the diagnosis systems used in test 2: used thresholds.

KPI	Thresholds
Retainability	[0.99, 0.997]
Accessibility	[0.992, 0.998]
Number of RRC Connections	[5846, 20703]
Number of ping-pong HO	[18, 83]
Number of bad cov. reports	[217, 1070]
CPU average load [%]	[22.5, 34.45]

4.2.3. Results

Once the standalone diagnosis systems have been trained, the performance metrics DER, FPR, FNR and OER have been computed for both the standalone diagnosis systems and the rules described in Table 2. In this test, the modeling-to-testing ratio has been varied from 10% to 90% in steps of 10, making 10 iterations per step. As in Section 4.1.3, a random permutation of the cases used for modeling and testing has been done in each of these 10 iterations. The resulting metrics have been then averaged. Table 12 shows the metrics that result of using a proportion of 60% in the modeling-to-testing ratio. This ratio has proved to minimize the values of all the metrics in this test. Unlike in Section 4.1, this scenario is made from real cases and contains outliers, that is, atypical cases. As the modeling-to-testing ratio rises, the probability for these outliers to belong to the modeling set also rises, thus inducing the behavior models to deviate from modeling the trend of the typical cases given a fault cause. On the other hand, if no outliers are taken into account during the model-fitting procedure their fault cause will not be predictable in the second stage and the error rates will also rise up.

As it can be seen in Table 12, in most cases, the combined diagnosis system outperforms the standalone diagnosis systems. Concretely, the median rule achieves the lowest overall error rate with a 5.39%, approximately 2/3 from that of the best standalone diagnosis system. However, the most relevant improvement takes place in the reduction of the FNR, which has been reduced a 46%. The FNR gives an idea of the amount of problematic causes wrongly deemed as normal. It is crucial making this metric as low as possible, since considering a problematic case as normal may result

Table 11
Diagnosis models for the diagnosis systems used in test 2: used rules.

Diagnosis model 1: from genetic algorithm							Diagnosis model 2: from data driven algorithm						
KPI						Diagnosis	KPI						Diagnosis
H	H	–	–	L	L	1	H	H	L	–	L	L	1
H	–	H	H	L	L	1	H	H	–	H	L	L	1
–	H	H	L	–	L	2	H	–	H	H	L	L	1
H	–	–	L	–	H	2	–	L	H	L	–	H	2
H	–	–	L	H	–	2	–	H	H	–	H	H	2
H	–	H	L	–	–	2	–	–	H	L	H	H	2
–	L	H	L	–	H	2	L	–	H	–	H	H	2
H	H	–	–	H	–	2	H	H	H	–	H	–	2
–	H	H	–	H	–	2	H	L	–	L	L	H	2
–	–	H	L	H	–	2	H	–	H	L	–	H	2
H	–	H	–	H	H	2	H	–	H	L	H	–	2
H	–	H	–	H	H	2	–	L	L	L	L	L	3
–	L	L	L	L	L	3	L	L	L	L	H	–	4
L	L	–	–	H	H	4	–	L	L	L	H	H	4
L	L	L	–	H	–	4	L	L	L	–	H	H	4
L	L	H	L	L	L	4	L	–	L	L	H	H	4
L	L	L	–	–	H	4							
L	–	L	–	H	H	4							
–	–	L	L	H	H	4							
L	L	L	H	–	–	4							

Table 12
Results of test 2: Combining two different algorithms.

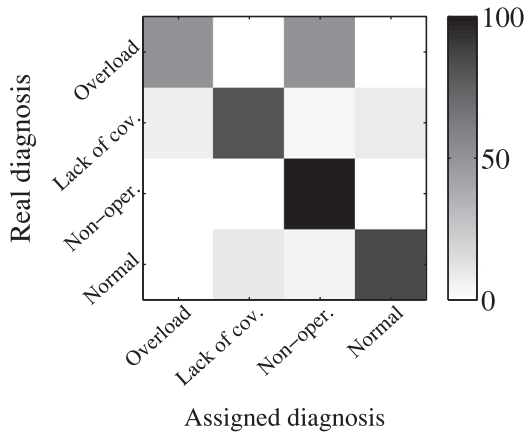
	DER	FPR	FNR	OER
Training: Data driven algorithm	2.62%	16.91%	6.47%	11.43%
Training: Genetic algorithm	1.87%	16.61%	2.68%	8.16%
Ensemble method				
Product rule	2.6%	12.21%	1.32%	6.2%
Sum rule	**1.78%**	11.55%	**1.25%**	5.59%
Max rule	**1.78%**	11.51%	**1.25%**	5.57%
Min rule	2.05%	11.42%	1.4%	5.84%
Median rule	**1.78%**	**10.67%**	1.34%	**5.39%**
Majority vote rule	**1.78%**	11.23%	**1.25%**	5.49%

in the worst case in unnoticed service outages and degradation in the network performance. Regarding this, the proposed method has proved to successfully reduce the FNR. Other indicators are not as critical. For example, misleading a fault cause with another may be to some extent tolerable (DER); although the actual problem is not that one the operator thinks it is, he is still aware of a problem in the network. Even considering normal cases as faulty may be tolerable as the network performance is not really degraded (FPR).

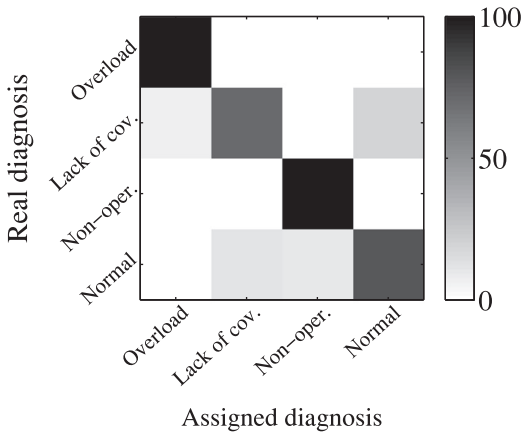
These results can also be seen in the normalized confusion matrices from the diagnosis methods. Fig. 6a shows the normalized confusion matrix for the FLC using genetic algorithm for rule learning; Fig. 6b shows the confusion matrix given the *data driven* algorithm was used for learning the rules and Fig. 6c shows the matrix from applying the median rule with a 60% of modeling-to-testing ratio in the ensemble method. In these matrices, the elements from the fourth column (excluding the main diagonal) account for the false negatives and the elements from the fourth row account for the false negatives. It can be seen how the elements from the main diagonal are reinforced in the ensemble method and how only those diagnoses which are mistaken by both baseline systems are slightly inherited by the latter. Fig. 6c also shows graphically how the FPR and FNR dropped with respect to those from the standalone systems.

5. Future lines of work

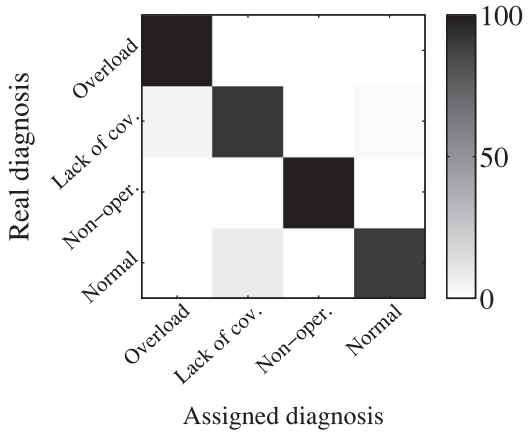
- *Decision templates*. The proposed method does not punish or reward the classifiers according to its performance during the training stage. Going a step further from the idea of the weighted majority vote used in Wei et al. (2014), a score system based on class and classifier aware decision templates applied over the a posteriori probabilities $P(w_m^r|\bar{x})$ from Eq. (4) could be used to improve the overall accuracy.
- *Non-parametric PDFs*. The proposal of analytically and parameter-defined PDFs results in a really light way of representing a statistical behavior, as only its parameters must be stored Table 1 to model a diagnosis system. However, these distributions may limit to some extent the statistical representation of the features from the training cases and they may eventually introduce a source of error in the posterior computation of $P(w_m^r)$ in case these cases follow a distribution that has not been considered. To solve this, the future research could focus on using non-parametric PDFs, like the kernel-based ones.
  Probability density functions may be classified into parametric and non-parametric functions. The former have analytic expressions and their shape depends on the parameters those functions hold. The latter, however, are defined by means of a kernel function. If all the cases from the dataset are placed along the axis given by a feature of interest (a certain KPI, for example) and a kernel function is centered wherever a point is, an empirical non-parametric PDF would result from averaging the sum of these functions over the number of cases. The main advantage of this method is its accuracy when modeling an empirical distribution. Its main drawback is that, since it is not defined by any parameters, it should be computed and stored point by point, possibly increasing the storage and computing requirements. This method, however, may be used together with (c). First, a reduced set of synthetic KPIs is computed and then, their PDFs are accurately estimated with this method.
- *Use of synthetic KPIs via feature extraction.* As it is described in Section 3.1, $N^r \times M^* \times R$ PDFs should be estimated in order to model all the feature-class-classifier relations. If any of these factors is relatively high, the computing cost for all these PDFs to be computed could be prohibitive. Due to this, working with a reduced group of synthetic/extracted features is proposed in an attempt of mapping the N original features into $\hat{N}$ synthetic features with $\hat{N} < N$.



(a) Normalized confusion matrix of the baseline diagnosis system with the genetic algorithm for rule learning.



(b) Normalized confusion matrix of the baseline diagnosis system with the *data driven* algorithm for rule learning.



(c) Normalized confusion matrix of the proposed method for combination using the median rule and a modeling-to-testing ratio of 60%.

Fig. 6. Normalized confusion matrices for the second test.

In the recent years and mainly motivated by the impulse of data mining many methods for dimensionality reduction have arisen. Within these, it is worth highlighting the Principal Component Analysis method (PCA) (Jolliffe, 2002). In an N-dimensional vector space, the simplest version of PCA (linear PCA) is a technique that finds the mutually-uncorrelated vectors onto which the projection of the samples generates the highest variances. The result is a set of orthogonal vectors sorted in descending order of achieved variance. The first of these vectors is that onto which the variance of the projection of the samples is maximum. In this sense, the original KPIs constitute the N-dimensional vector space basis, whereas the $\hat{N}$ synthetic KPIs represent the orthogonal vectors with the highest variance. To be rigorous, up to N synthetic orthogonal KPIs may be computed. However, only a small set of them, the first $\hat{N}$, is enough to account for most of the variance of the data.
By applying this technique, based on the eigenvalue decomposition of the covariance matrix of the original KPIs, these can be mapped into $\hat{N}$, preserving most of the information contained in the former.

6. Conclusions

A hybrid ensemble of classifiers, devised to merge expert knowledge from different sources has been presented and assessed in the context of fault cause diagnosis in cellular networks, allowing the expertise from several troubleshooting experts and the knowledge contained in databases of cases previously diagnosed to be combined in order to develop a more accurate diagnosis system.

Unlike the common approach of hybrid ensembles, based on the majority vote of their baseline components, this work proposes a hybrid ensemble of classifiers obtained from the combination of the statistical behavior models of the baseline diagnosis systems. This approach allows obtaining and afterwards combining by just applying some algebraic rules the partial diagnoses from the standalone classifiers without actually needing them to assess every case under test, thus reducing the computational cost of usual hybrid ensembles of classifiers.

The method has been tested with two different sources of cases under test: cases provided by an LTE RAN simulator and cases gathered from a real live LTE network. Likewise, two use cases have been assessed: the combination of diagnosis models designed by two different network troubleshooting experts and the combination of two diagnosis systems using different learning algorithms. The proposed method has proved to outperform the behavior of its base components in both tests in terms of the diagnosis error rate, proving to be an effective tool in the fault cause diagnosis in current and future self-healing networks.

Acknowledgment

This work has been partially funded by Optimi-Ericsson, Junta de Andalucía (Consejería de Ciencia, Innovación y Empresa, Ref. 59288 and Proyecto de Investigación de Excelencia P12-TIC-2905) and ERDF.

References

3GPP (a). Evolved Universal Terrestrial Radio Access (E-UTRA) Radio Resource Control (RRC); Protocol Specification, Rel-13, Version 13.2.0, (2015-12). TS 36.331. 3rd Generation Partnership Project.

3GPP (b). Feasibility study for Further Advancements for E-UTRA (LTE-Advanced), Rel-13, Version 13.0.0 (2015–12). TR 36.912. 3rd Generation Partnership Project.

3GPP (c) (May 2004). OFDM-HSDPA System level simulator calibration (R1-040500). *3GPP TSG-RAN WG1 37*. 3rd Generation Partnership Project (3GPP).

3GPP (d). Self-Organizing Networks (SON); Concepts and requirements, Rel-13, Version 13.0.0 (2015–12). TS 32.500. 3rd Generation Partnership Project.

3GPP (e). Self-Organizing Networks (SON); Self-Healing concepts and requirements, Rel-13, Version 13.0.0 (2015–12). TS 32.541. 3rd Generation Partnership Project.

Barco, R., Díez, L., Wille, V., & Lázaro, P. (2009). Automatic diagnosis of mobile communication networks under imprecise parameters. *Expert Systems with Applications, 36*(1), 489–500. doi:10.1016/j.eswa.2007.09.030.

Barco, R., Lazaro, P., Diez, L., & Wille, V. (2008). Continuous versus discrete model in autodiagnosis systems for wireless networks. *IEEE Transactions on Mobile Computing, 7*(6), 673–681. doi:10.1109/TMC.2008.23.

Barco, R., Lázaro, P., Wille, V., Díez, L., & Patel, S. (2009). Knowledge acquisition for diagnosis model in wireless networks. *Expert Systems with Applications, 36*(3), 4745–4752. doi:10.1016/j.eswa.2008.06.042.

Barco, R., Lázaro, P., & Muñoz, P. (2012). A unified framework for Self-Healing in wireless networks. *IEEE Communications Magazine, 50*(12), 134–142. doi:10.1109/MCOM.2012.6384463.

Begum, S., Chakraborty, D., & Sarkar, R. (2015). Cancer classification from gene expression based microarray data using SVM ensemble. In *2015 international conference on condition assessment techniques in electrical systems (CATCON)* (pp. 13–16). doi:10.1109/CATCON.2015.7449500.

Breiman, L. (1996). Bagging predictors. In *Machine learning* (pp. 123–140).

Ciocarlie, G., Lindqvist, U., Nováczki, S., & Sanneck, H. (2013). Detecting anomalies in cellular networks using an ensemble method. In *Proceedings of the 9th international conference on network and service management (CNSM 2013)* (pp. 171–174). doi:10.1109/CNSM.2013.6727831.

Dasarathy, B., & Sheela, B. V. (1979). A composite classifier system design: concepts and methodology. *Proceedings of the IEEE, 67*(5), 708–713. doi:10.1109/PROC.1979.11321.

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119–139. doi:10.1006/jcss.1997.1504.

Gandhi, I., & Pandey, M. (2015). Hybrid ensemble of classifiers using voting. In *2015 international conference on green computing and internet of things (ICGCIoT)* (pp. 399–404). doi:10.1109/ICGCIoT.2015.7380496.

Gómez-Andrades, A., Muñoz Luengo, P., Khatib, E., de la Bandera Cascales, I., Serrano, I., & Barco, R. (2015). Methodology for the design and evaluation of Self-Healing LTE networks. *IEEE Transactions on Vehicular Technology, PP*(99). doi:10.1109/TVT.2015.2477945. 1–1

Gómez-Andrades, A., Muñoz, P., Serrano, I., & Barco, R. (2016). Automatic root cause analysis for LTE networks based on unsupervised techniques. *IEEE Transactions on Vehicular Technology, 65*(4), 2369–2386. doi:10.1109/TVT.2015.2431742.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., & Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computing, 3*(1), 79–87. doi:10.1162/neco.1991.3.1.79.

Jolliffe, I. (2002). Principal component analysis. *Springer Series in Statistics* (2nd). Springer-Verlag New York.

Khatib, E. J., Barco, R., Gómez-Andrades, A., Muñoz, P., & Serrano, I. (2015). Data mining for fuzzy diagnosis systems in LTE networks. *Expert Systems with Applications, 42*(21), 7549–7559. doi:10.1016/j.eswa.2015.05.031.

Khatib, E. J., Barco, R., Gómez-Andrades, A., & Serrano, I. (2015). Diagnosis based on genetic fuzzy algorithms for LTE Self-Healing. *IEEE Transactions on Vehicular Technology*. doi:10.1109/TVT.2015.2414296.

Kittler, J., Hatef, M., Duin, R. P. W., & Matas, J. (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(3), 226–239. doi:10.1109/34.667881.

Kuncheva, L. (2002). A theoretical study on six classifier fusion strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(2), 281–286. doi:10.1109/34.982906.

Liu, H., Chen, G., Song, G., & Han, T. (2009). Analog circuit fault diagnosis using bagging ensemble method with cross-validation. In *International conference on mechatronics and automation, 2009. ICMA 2009* (pp. 4430–4434). doi:10.1109/ICMA.2009.5246675.

Mehlführer, C., Wrulich, M., Colom Ikuno, J., Bosanska, D., & Rupp, M. (2009). Simulating the long term evolution physical layer. In *Proc. of 17th European signal processing conference (EUSIPCO)*.

Muñoz, P., de la Bandera, I., Ruíz, F., Luna-Ramírez, S., Barco, R., Toril, M., et al. (2011). Computationally-efficient design of a dynamic system-level LTE simulator. *International Journal of Electronics and Telecommunications, 57*(3), 347–358. doi:10.1155/2012/802606.

Nováczki, S. (2013). An improved anomaly detection and diagnosis framework for mobile network operators. In *2013 9th international conference on the design of reliable communication networks (drcn)* (pp. 234–241).

Shen, H.-B., & Chou, K.-C. (2006). Ensemble classifier for protein fold pattern recognition. *Bioinformatics, 22*(14), 1717–1722. doi:10.1093/bioinformatics/btl170.

Szilágyi, P., & Nováczki, S. (2012). An automatic detection and diagnosis framework for mobile communication systems. *IEEE Transactions on Network and Service Management, 9*(2), 184–197. doi:10.1109/TNSM.2012.031912.110155.

Wei, H., Lin, X., Xu, X., Li, L., Zhang, W., & Wang, X. (2014). A novel ensemble classifier based on multiple diverse classification methods. In *2014 11th international conference on fuzzy systems and knowledge discovery (FSKD)* (pp. 301–305). doi:10.1109/FSKD.2014.6980850.

Wiezbicki, T., & Ribeiro, E. P. (2016). Sensor drift compensation using weighted neural networks. In *2016 IEEE conference on evolving and adaptive intelligent systems (EAIS)* (pp. 92–97). doi:10.1109/EAIS.2016.7502497.

Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., et al. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems, 14*(1), 1–37. doi:10.1007/s10115-007-0114-2.

Yuksel, S., Wilson, J., & Gader, P. (2012). Twenty years of mixture of experts. *IEEE Transactions on Neural Networks and Learning Systems, 23*(8), 1177–1193. doi:10.1109/TNNLS.2012.2200299.