(19) Евразийское патентное ведомство

(21) 202392565 (13) A1

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОЙ ЗАЯВКЕ

(43) Дата публикации заявки
2023.11.02

(22) Дата подачи заявки
2022.03.15

(51) Int. Cl. ***G05B 13/02*** (2006.01)
G05B 23/02 (2006.01)
G05B 19/418 (2006.01)

(54) СОЗДАНИЕ ВИРТУАЛЬНЫХ ДАТЧИКОВ ДЛЯ ИСПОЛЬЗОВАНИЯ В ПРОИЗВОДСТВЕННЫХ УСТАНОВКАХ

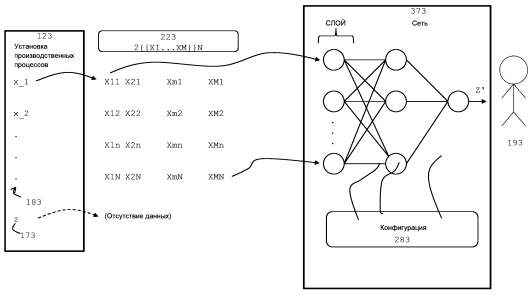
(31) **LU102672**
(32) **2021.03.16**
(33) **LU**
(86) **PCT/EP2022/056710**
(87) **WO 2022/194871 2022.09.22**

(71) Заявитель:
ПАУЛЬ ВЮРТ С.А. (LU)

(72) Изобретатель:
Соккарт Седрик (LU)

(74) Представитель:
Веселицкий М.Б., Кузенкова Н.В., Каксис Р.А., Белоусов Ю.В., Куликов А.В., Кузнецова Е.В., Соколов Р.А., Кузнецова Т.В. (RU)

(57) Производственная установка (123) может не иметь датчика для конкретного параметра, так что компьютер использует нейронную сеть (373) для виртуализации отсутствующего датчика. Компьютер обучает нейронную сеть (373) для получения индикатора (Z') параметра для другого параметра (173, z) технологического процесса производственной установки (123) с шагами, которое включают в себя прием содержащих исторические данные измерений временных рядов измерений от базовых установок, получение правил преобразования путем обработки временных рядов в ряды признаков, которые являются инвариантными к относящимся к областям значений различиям базовых установок, преобразование временных рядов посредством правил преобразования, прием одномерного временного ряда измерений другого параметра (z) технологического процесса и обучение нейронной сети с использованием ряда признаков на входе и с использованием одномерного временного ряда измерений на выходе.



202392565 A1

202392565 A1

СОЗДАНИЕ ВИРТУАЛЬНЫХ ДАТЧИКОВ ДЛЯ ИСПОЛЬЗОВАНИЯ В ПРОИЗВОДСТВЕННЫХ УСТАНОВКАХ

Область техники

В целом, раскрытие относится к процессам промышленного производства, и, прежде всего, раскрытие относится к компьютерным системам, способам и программным продуктам для эмуляции датчиков и эмуляции данных измерений технологических параметров осуществляющих эти процессы производственных установок.

Уровень техники

Эксплуатация производственной установки и одновременное измерение ее технологических параметров неразрывно связаны друг с другом. Измерение включает в себя измерение параметров, которые непосредственно связаны с протекающим в установке процессом, например, измерение температуры, давления, звука, излучения (по показаниям инфракрасной камеры), отражения света от поверхностей, концентрации газов и других физических свойств или явлений. Но измерение может относиться и к параметрам процесса, которые косвенно связаны с ним, например, измерение физических свойств поступающих в установку материалов или выходящих из установки материалов (результат процесса).

В этом смысле производственная установка является установкой под наблюдением.

Кроме того, параметры технологического процесса могут быть разграничены по местам внутри установки за ее пределами. Например, распределение температуры может быть измерено в разных частях установки.

В более общем виде производственная установка может представлять собой оборудование, выполняющее какой-либо производственный процесс. Например, это может быть химический реактор, металлургическая печь, технологическая емкость, двигатель. Более конкретно, в аспекте принципа действия, печь может представлять собой доменную печь. Сильно упрощая, в доменную печь поступают руда и кокс, а также горячий воздух (через фурмы), и выдается расплавленный металл.

Измерение является основным назначением связанных с установкой датчиков. Например, установка может быть оснащена температурными датчиками. В качестве таких датчиков могут выступать термометры, резистивные температурные датчики типа PT100 или другие типы температурных датчиков.

Известно, что результаты измерений могут быть предоставлены по меньшей мере двумя видами измерений:

- посредством связанных с установкой датчиков, и
- посредством взаимодействия с установкой человека-оператора.

В примере с доменной печью датчики распределены в различных местах, кроме того, оператор обычно берет пробы расплавленного материала. Упрощая, оператор открывает часть горна доменной печи для извлечения расплавленного материала (при применении сверлильной бурмашины этот шаг называют пробиванием). Затем материал поступает в желоб (называют заливкой металла в форму), и оператор получает возможность проверки образцов на химический состав, температуру, внешний вид и тому подобное.

Большинство результатов измерений преобразуют в данные измерений. Относительно видов данных, измерительные данные могут иметь различные по сложности форматы, например:

- скаляры для представления температуры или давления,
- векторы для представления траекторий движения деталей установки,
- матрицы для представления образов с поверхностей и тому подобного.

Данные измерений обрабатывают посредством одного или нескольких компьютеров. Компьютеры могут обрабатывать данные измерений для получения индикаторов рабочего состояния установки. На основе данных измерений и показателей оператор может изменять рабочие параметры. Например, упомянутое взятие проб может выявить недостаточную температуру. Совместно с другими данными компьютер указывает оператору на необходимость загрузки в печь большего количества кокса (количество кокса является рабочим параметром).

Во многих других сценариях компьютер может работать непосредственно с установкой в функции устройства управления процессом (например, цикла управления температурным параметром).

Однако данные измерений не всегда являются доступными по ряду причин и ограничений. Некоторые из них заслуживают дополнительного внимания:

- Возможная поломка конкретного датчика или нарушение линии передачи данных к компьютеру.
- Для различных модификаций установки распределение датчиков может отличаться от установки к установке, также когда установки являются в основном однотипными. Если продолжить пример с печами, то их строят по индивидуальным проектам или очень малыми сериями. Таким образом, парк печей является неоднородным. Поэтому печи с одинаковым принципом работы (например, терморегулирование путем загрузки кокса и/или вдувания угольной пыли), одинаковым сырьем (например, железная руда, кокс), тем не менее, могут быть отличными (например, по объему). Такие различия между установками могут также проявляться в виде недостаточности данных измерений.
- Свое значение имеет и фактор взаимодействия между человеком и установкой. Имеется еще одна техническая проблема, связанная с потенциальной опасностью или риском для оператора. Открытие печи для взятия замеров должно быть ограничено несколькими случаями. Кроме того, некоторые измерения могут потребоваться во время критических операций с печью, например, измерение температуры расплавленного материала в процессе заливки металла в форму. Участие человека в измерениях должно быть сведено к минимуму таким образом, что оператор может полностью сосредоточиться на работе печи.
- Некоторые из этих ограничений могут быть проявлены в комбинации: устаревший аналоговый термометр требует оператора для считывания данных, цифровой термометр - нет.

Компьютеры могут вычислять индикаторы рабочего состояния установки, если известны соотношения между данными измерений. Например, компьютер в автомобиле может предложить водителю переключить передачу. Однако во многих случаях связь между данными измерений является слишком сложной для ее моделирования с помощью математических уравнений.

Решением проблемы является использование нейронных сетей. Сеть принимает данные измерений и выдает индикатор. Проще выражаясь, сеть состоит из так называемых нейронов, которые расположены во множестве слоев.

Взаимосвязь между нейронами регулируется так называемыми весами. Набор весов в сети может быть назван конфигурацией.

Однако сеть должна быть обучена (для настройки весов и других параметров сети). Опять-таки упрощая, обучение заключается в подаче в сеть данных измерений за прошлые периоды.

Однако исторические данные могут оказаться недоступными для конкретной установки, прежде всего в тех случаях, когда установки не имеют подходящих прототипов или аналогов. Отсутствие исторических данных является особо характерным для установок (например, печей), которые производятся в различных модификациях.

Дополнительную справочную информацию можно получить из следующих публикаций:

Li Xiang et al: «Diagnosing Rotating Machines With Weakly Supervised Data Using Deep Transfer Learning», IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE SERVICE CENTER, NEW YORK, NY, US, vol. 16, no. 3, 8 July 2019 (2019-07-08), pages 1688-1697. В публикации говорится о диагностике оборудования на основе данных. Имеется проблема отсутствия в достаточном количестве обучающих данных для установок в целевой области значений. Однако вместо них могут быть использованы обучающие данные из нескольких исходных областей значений. В публикации описан подход к приспосабливанию обучающих данных путем переноса обучения.

Li Xiang et al: «Domain generalization in rotating machinery fault diagnostics using deep neural networks», NEUROCOMPUTING, ELSEVIER, AMSTERDAM, NL, vol. 403, 8 May 2020 (2020-05-08), pages 409-420. В публикации также описана диагностика оборудования на основе данных. Имеется задача извлечения признаков, которые являются общими или инвариантными по отношению к областям значений. В публикации описан подход с использованием состязательной сети области значений.

Liao Yixiao et al: «Deep Semisupervised Domain Generalization Network for Rotary Machinery Fault Diagnosis Under Variable Speed», IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, IEEE, USA, vol. 69, no. 10, 6 May 2020 (2020-05-06), pages 8064-8075. В публикации рассмотрено вращающееся оборудование и использование нейронных сетей для обнаружения неисправностей. Однако для одной скорости вращения маркированные

обучающие данные могут присутствовать, а для другой скорости - нет. Тем не менее, использование порождающей состязательной сети Вассерштейна позволяет решить эту проблему.

Краткое описание изобретения

Работа производственной установки характеризуется несколькими параметрами технологического процесса, однако в ней может отсутствовать соответствующий датчик конкретного параметра, или данные измерений этого параметра технологического процесса могут оказаться недоступными по другим причинам. Нейронная сеть предоставляет индикатор параметра, который соответствует параметру технологического процесса в установке. Сеть обучена на основе полученных от базовых (референтных) установок исторических данных. По меньшей мере на одной из базовых установок, а именно на исходной базовой модели, имеется датчик данного параметра технологического процесса.

Базовые установки являются различными, и в аспекте машинного обучения они относятся к разным областям значений. В связи с этим при обучении используют перенос обучения, а точнее неконтролируемое приспосабливание к области значений.

В результате выполнения реализованного на компьютере способа компьютер обучает нейронную сеть, что позволяет ей в дальнейшем обрабатывать содержащие данные измерений многомерные временные ряды измерений. Данные измерений представляют собой отдельные параметры технологического процесса производственной установки. Обученная нейронная сеть настроена на предоставление индикатора параметра другого технологического параметра производственной установки.

На первом приемном шаге компьютер принимает первый содержащий исторические данные измерений многомерный временной ряд измерений от первой базовой установки принимает второй содержащий исторические данные измерений многомерный временной ряд измерений от физически другой второй базовой установки.

На шаге получения компьютер путем обработки первого и второго многомерных временных рядов измерений получает набор правил преобразования так, что правила преобразования позволяют

преобразовательному модулю преобразовывать соответственно первый и второй многомерные временные ряды измерений в первый и второй многомерные временные ряды признаков. Многомерные временные ряды признаков являются инвариантными к различиям областей значений первой и второй базовых установок.

На шаге преобразования компьютер преобразует первый многомерный временной ряд измерений в первый многомерный временной ряд признаков посредством применяющего правила преобразования преобразовательного модуля.

На втором приемном шаге компьютер принимает от первой базовой установки содержащий данные измерений другого параметра технологического процесса одномерный временной ряд измерений.

На шаге обучения компьютер обучает нейронную сеть, используя на входе сети первый многомерный временной ряд признаков, а на выходе сети - одномерный временной ряд измерений.

Факультативно, компьютер может получать набор правил преобразования путем выполнения следующих итеративных действий: преобразования с использованием набора предварительных правил преобразования первого многомерного временного ряда измерений в первый многомерный временной ряд признаков и преобразования второго многомерного временного ряда измерений во второй многомерный временной ряд признаков, различения посредством дискриминирующего модуля происхождения первого и второго временных рядов признаков как первого и второго многомерных временных рядов измерений, изменения набора предварительных правил преобразования до тех пор, когда дискриминатор более не может различать происхождение данных.

Факультативно, в компьютере может быть использован дискриминирующий модуль, который определяет, что более не может различать происхождение данных путем работы до тех пор, когда состязательные потери достигнут максимального значения.

Факультативно, компьютер может итеративно выполнять шаг получения правил преобразования и шаг обучения, причем преобразовательный модуль дополнительно принимает создаваемые нейронной сетью в процессе обучения сетевые потери так, что итерации останавливаются при минимальных сетевых потерях.

Факультативно, комбинация преобразовательного модуля и дискриминирующего модуля работает как порождающая состязательная сеть.

Факультативно, шаги способа, такие как прием многомерных временных рядов измерений, получение набора правил преобразования, преобразование и обучение нейронной сети компьютер выполняет для многомерных временных рядов измерений, для которых посредством модуля селекции данных переменные были определены как связанные с другим параметром технологического процесса.

Факультативно, модуль селекции данных обучают посредством модуля обучения селектора, который побуждает компьютер к итеративному выполнению шагов способа по получению, преобразованию и обучению, при этом критерием является минимизация функции потерь в процессе обучения нейронной сети.

Факультативно, компьютер выполняет прием содержащих исторические данные первого и второго многомерных временных рядов измерений после приспосабливания вида данных для данных измерений.

Факультативно, компьютер может приспосабливать вид данных посредством, по меньшей мере, одного из следующих действий: анализа содержащих образцы звука данных измерений путем классификации образцов звука по категориям и анализа содержащих образы данных измерений путем классификации образов по категориям.

Факультативно, компьютер итеративно выполняет приспосабливание для различных целей, причем цели выбирают по критерию минимизации функции потерь в процессе обучения нейронной сети.

Факультативно, компьютер использует нейронную сеть, которая является регрессионной нейронной сетью. Факультативно, данные измерений связаны с параметрами технологического процесса, которые представляют собой физические свойства, выбранные из следующих: температура, давление, химический состав материалов внутри установки, внешний вид части производственной установки, и звук, издаваемый производственной установкой.

Факультативно, производственные установки могут быть представлены химическими реакторами, металлургическими печами, технологическими емкостями, и двигателями.

После обучения нейронной сети компьютер (или физически другой компьютер) может выполнять реализованный на компьютере способ использования нейронной сети, включающий в себя прием содержащего данные измерений многомерного временного ряда измерений для наблюдаемой производственной установки, преобразование многомерного временного ряда измерений в многомерный временной ряд признаков и использования сети для предоставления индикатора параметра.

Факультативно, использование нейронной сети происходит в ситуациях, выбранных из следующего: отказ датчика, обрыв линии передачи данных, минимизация участия человека при измерении.

Предложен компьютерный программный продукт, который, будучи загруженным в память компьютерной системы и исполняемым посредством по меньшей мере одного процессора компьютерной системы, побуждает компьютерную систему к выполнению шагов компьютерно-реализуемого способа.

Предложена компьютерная система, которая включает в себя несколько компьютерно-реализуемых модулей, которые при выполнении компьютерной системой могут выполнять шаги компьютерно-реализуемого способа. Или с другой точки зрения: компьютерная система выполняет шаги компьютерно-реализуемого способа.

Кроме того, предложено использование выполняющего способ компьютера, который действует в качестве виртуального датчика для получения индикатора параметра, выступающего в роли параметра производственной установки.

Краткое описание чертежей

Далее подробно описаны варианты осуществления настоящего изобретения с отсылками на прилагаемые чертежи, на которых:

Фиг. 1 для пояснения применяемых терминов показывает производственную установку совместно с многомерными временными рядами измерений, а также нейронную сеть совместно с конфигурацией,

Фиг. 2 показывает производственные установки компьютерные модули в фазе подготовки к получению набора правил преобразования,

Фиг. 3 показывает базовую производственную установку, преобразовательный модуль и обучаемую сеть в фазе обучения,

Фиг. 4 показывает производственную установку на фиг. 1 и преобразовательный модуль на фиг. 3 в рабочей фазе для предоставления индикатора параметра,

Фиг. 5 показывает блок-схемы способа обучения нейронной сети способа использования сети в процессе работы производственной установки,

Фиг. 6 показывает факультативный модуль адаптера вида, реализующий факультативное видовое приспосабливание данных измерений,

Фиг. 7 показывает факультативный модуль селектора данных, который факультативно реализует выбор относящихся к делу данных измерений,

Фиг. 8 показывает пример использования производственных установок, в качестве которых выступают доменные печи, и

Фиг. 9 показывает пример типового компьютерного устройства, которое может быть использовано в описанных в настоящем документе способах.

Подробное описание вариантов осуществления

Обзор применяемых терминов

Фиг. 1 для пояснения применяемых терминов показывает производственную установку 123 совместно с многомерными временными рядами 223 измерений, а также нейронную сеть 373 совместно с конфигурацией 283,

На чертеже также показано (хотя и символически), что производственная установка 123 осуществляет производственный процесс с параметрами 173/183 технологического процесса. Параметры технологического процесса могут быть подразделены на две группы:

- измеряемые параметры 183 (или x-параметры) и
- неизмеряемый параметр 173 (или z-параметр, по меньшей мере один).

Определяющий признак «измерение» связан с конкретной производственной установкой 123 в том смысле, что данные измерений попадают в многомерный временной ряд 223 измерений для этой установки 123.

По меньшей мере, для одного параметра z технологического процесса данные измерений отсутствуют (то есть, являются недоступными). В принципе, параметр z является измеряемым параметром, но данные о нем недоступны. На выходе нейронной сети 373 получают индикатор Z' параметра, который является индикатором данного конкретного параметра z технологического процесса. Индикатор Z' параметра можно рассматривать как «искусственное значение измерения» или как искусственные или виртуальные данные измерения. Иными словами, компьютер эмулирует датчик.

Индикатор Z' параметра может служить индикатором состояния установки 123 (Z' указывает на состояние), либо Z' может служить входной переменной для компьютера или другого модуля, определяющего состояние установки (Z' является одним из влияющих на состояние факторов). В данном описании основное внимание уделено вычислению Z'.

Для оператора (человека) 193 производственной установки 123 сеть 373 представляется виртуальным датчиком, который выводит индикатор Z' параметра. В теоретических и идеальных ситуациях индикатор Z' представляет физическую реальность, как это может делать реальный датчик (аппаратный датчик, физический датчик).

В практических, неидеальных ситуациях точность индикатора Z' параметра (вычисляемого посредством сети 373) может быть определена как различие между индикатором Z' от сети 373 и данными Z измерений от соответствующего аппаратного датчика.

В данном описании пояснен подход к определению Z' (фиг. 1-5) и описаны факультативные мероприятия по повышению точности в связи с фиг. 6-7.

Производственная установка 123 (или производственное оборудование) является установкой, которая непрерывно осуществляет производственные процессы. Хотя в описании уже были упомянуты некоторые примеры, основное внимание в нем уделено примеру, когда установка 123 является доменной печью.

В примере с доменной печью параметр z технологического процесса может быть представлен температурой расплавленного материала. Отсутствует получаемое в результате прямого измерения значение (нет датчика температуры или имеется другая упомянутая выше причина), но индикатор Z' параметра предоставляет расчетное значение. Параметр Z' может быть, кроме того,

преобразован в индикатор состояния (например, если печь готова к разливке, поскольку температура и другие критерии позволяют это сделать).

Для обеспечения сети 373 возможности предоставления Z', некоторые шаги способа должны быть выполнены заранее, включая сюда машинное обучение. Для удобства в описании шаги описаны в контексте фаз. В описании могут различаться компоненты, которые являются активными на подготовительной фазе **1 (см. фиг. 2), на обучающей фазе **2 (см. фиг. 3) и на рабочей фазе **3 (см. фиг. 4). Другими словами, в данном описании ссылки, обозначенные как **1/**2/**3, означают компоненты, которые являются одинаковыми, но различаются в этих фазах. Например, установка 121 на фиг. 2 и установка 123 на фиг. 4 физически являются одной и той же установкой, которая работает при этом как базовая установка (или как установка под наблюдением/контролем).

В описании фазы **1/**2/**3 различаются в связи с сетью, которая предоставляет индикатор Z' параметра. С этой точки зрения, обучение происходит в фазе **2.

Однако могут быть и другие обучающие циклы:

- Подготовительная фаза **1 может факультативно включать в себя обучающий цикл (см. фиг. 5 с циклом итеративного выполнения подшагов получения правил 421).
- Обучение может быть применено и для видового приспосабливания (см. фиг. 6), при применении циклов для **1 и **2.
- Другой обучающий цикл может быть применен при более крупном временном разбиении, при котором итеративно выполняются некоторые шаги из **1 и **2 (см. фиг. 7).
- Еще один обучающий цикл может быть применен для обучения дискриминатора и преобразователя при введении в него в качестве параметра сетевых потерь LOSS.

Измеряемые (или, по меньшей мере, эмулируемые) параметры технологического процесса обозначаются строчными буквами, например, «x» или «z» (приведены слева в пределах установки 123).

На чертеже это показано для параметров x_1 (это может быть температура в определенном месте внутри установки, например, в верхней части) и x_2 (например, температура в другом месте, в нижней части). Имеется N параметров

технологического процесса {x_n}N и по меньшей мере один другой параметр z. Нижнее подчеркивание является местозаполнителем.

Данные измерений обозначаются заглавными буквами, например, X или Z. Индикатор Z' параметра обозначается символом штрих ('), поскольку он обозначает либо фактические данные измерений, либо искусственные данные измерений.

Заглавная буква применяется и для промежуточных данных «Y» (или «данных признаков», которые не подлежат измерению и не являются выходными данными).

Временные ряды

Данные (данные измерений, промежуточные данные) могут быть доступными в виде временных рядов, то есть рядов значений данных, индексированных во временном порядке для последовательных моментов времени.

На чертеже временные ряды представлены в виде краткого обозначения (прямоугольник 223) и в виде матрицы.

Обозначение {X1... XM} представляет единичный (то есть, одномерный) временной ряд с элементами данных измерений Xm (или кратко «элементами»). Элементы Xm имеются от момента 1 времени до момента M времени: X1, X2,..., Xm, ...XM (то есть, «временной ряд измерений»). Индекс m является индексом момента времени. За моментом m времени следует момент (m+1) времени, который обычно удален на одинаковый интервал Δt.

В качестве примера можно привести повышение температуры за M моментов времени: {1400°C ... 1500°C}. Специалисты в данной области техники могут предварительно обрабатывать значения данных, например, до нормализованных значений [0,1], или {0,2 ... 1}. Формат данных измерений не ограничивается скалярами или векторами, {X1... XM} может также обозначать последовательность из M образов, полученных от момента 1 времени до момента M времени. В случае, когда оборудование является доменной печью, образы могут быть получены изнутри фурм печи или из непосредственного окружения доменной печи. Образы снимают для выявления аномалий.

Обозначение {{X1... XM}}N представляет многомерный временной ряд (на фиг. 1 условное обозначение 223 над матрицей), содержащий в качестве элементов данных векторы {X_m}N от момента 1 времени до момента M времени. Векторы имеют кардинальное число N (число переменных, то есть параметров технологического процесса, для которых имеются данные измерений), то есть в любой момент времени от 1 до M имеется N элементов данных. В матрице в качестве индекса строки указан индекс n переменной (от x_1 до x_N).

Например, одиночный временной ряд для температуры может сопровождаться одиночным временным рядом для давления, для данных измерений химического состава материалов или тому подобным.

Специалист в данной области техники понимает, что описание является упрощенным. Реальные числа N переменных могут достигать нескольких тысяч и превышать это значение. Временные ряды не являются идеальными. Иногда какой-либо элемент отсутствует, но специалист может разрешить такие ситуации.

Выбор временного интервала Δt и числа моментов M времени зависит от процесса. Общая длительность $\Delta t*M$ временного ряда (то есть, размер окна) соответствует тому изменению параметров технологического процесса, которое занимает наибольшее время. Например, для обработки материалов в печи требуется несколько часов, поэтому $\Delta t*M$ выбирают соответствующим образом. Интервалы Δt удобно выбирать в диапазоне от 1 до 60 минут, а $\Delta t*M$ - в диапазоне от 4 часов до 24 часов. В примере с доменной печью можно применять $\Delta t = 15$ минут и $M = 32$, что соответствует 8-часовой рабочей смене.

Поскольку моменты tm времени задают время обработки данных посредством сети, некоторые данные измерений могут быть предварительно обработаны. Например, датчик температуры может предоставлять данные каждую минуту, но для $\Delta t=15$ минут часть данных может быть отброшена, усреднена по Δt или обработана иным образом.

Поскольку данные могут поступать от различных производственных установок, иногда перед {} или {{...}} указаны 1, 2 и тому подобное. В приведенном на чертеже примере 2{{...}} означает, что многомерный временной ряд поступает от установки 123.

Как будет пояснено, компьютер применяет преобразование X в Y. Полученные значения могут быть представлены и многомерными временными рядами. Кардинальные числа заданы величиной K.

Условные обозначения

В данном документе условные обозначения 1** указывают на производственную установку в реальном мире (физическом мире), условные обозначения 2** указывают на данные (например, конфигурацию 283), условные обозначения 3** указывают на аппаратные средства и компьютерно-реализуемые модули (например, нейронную сеть 373), а условные обозначения 4** или 5** указывают на шаги способа (см. фиг. 5).

Компьютеры

Обозначение «компьютер» (в единственном числе, без условного обозначения) означает вычислительную функцию или функцию компьютерно-реализуемого модуля (например, вычислительные блоки или разделенные по времени ресурсы таких блоков). Функции могут быть распределены между различными физическими компьютерами, при этом компьютерно-реализуемые модули распределены между такими различными компьютерами.

На чертежах также показана компьютерная программа или компьютерный программный продукт. Компьютерный программный продукт, будучи загруженным в память компьютера и исполняемым посредством по меньшей мере одного процессора компьютера, выполняет шаги компьютерно-реализуемого способа. Другими словами, программа содержит команды для модулей.

С другой стороны, на чертежах показаны модули компьютерной системы, включающей в себя несколько компьютерно-реализуемых модулей, которые при выполнении посредством компьютерной системы выполняют шаги компьютерно-реализуемого способа. Производственные установки не рассматриваются в качестве компьютерно-реализованных модулей.

Инструмент для машинного обучения и его конфигурация

На чертеже также изображена нейронная сеть 373, имеющая расположенные слоями (нейроны на одной вертикальной линии) нейроны (круговые символы). Нейронная сеть 373 представляет собой пример инструмента для машинного обучения. В некоторых других описанных в настоящем документе модулях также используется машинное обучение.

Существует два основных способа описания таких сетей:

- - набор гиперпараметров, и
- - конфигурация.

Гиперпараметры

- определяют подлежащие выполнению посредством нейронов математические функции (например, суммирование взвешенных входных сигналов и применение порогового значения или другие функции),
- определяют выполняемые посредством объединенных в комбинации нейронов функции (например, выбор функций ядра, между ядрами свертки, ядрами Гаусса и тому подобное),
- определяют расположения нейронов в слоях и взаимосвязи между слоями (например, полную связь между слоями, например, между первым и вторым слоями).

Гиперпараметры обычно описаны в уровне техники как архитектура сети. Обучение гиперпараметров не производится.

Конфигурация 283 определяет взвешенную передачу данных от нейрона к нейрону, а также задает значения смещения (или сдвига) для нейронов. Конфигурацию 283 получают в результате обучения.

Нейронная сеть 373 показана в данном случае только в качестве примера.

На фиг. 6-7 показаны другие инструменты (например, для приспосабливания вида данных, выбора относящихся к делу данных, преобразования данных), для которых соглашения по архитектуре и конфигурации также являются применимыми.

Относящиеся к областям значений различия (различия между областями значений) установок

Производственная установка 123 (фиг. 1) не является единственной доступной установкой для получения таких данных измерений. Для получения таких данных измерений являются доступными другие производственные установки, далее «аналогичные установки». Аналогичные установки работают по одному и тому же принципу, но могут отличаться по некоторым параметрам.

С учетом этих различий производственные установки можно рассматривать как принадлежащие к первой и второй областям значений.

Одна из аналогичных установок может предоставлять дополнительные данные измерений для обучения сети. В описании установки 111/112 названы

исходными (предоставляют реальные данные Z измерений для параметра z), а установка 123 - целевой (для нее сеть вычисляет показатель Z'). Предоставляющую данные для обучения аналогичную установку также называют «базовой установкой». Как исходная, так и целевая установки являются базовыми в том смысле, что обе могут предоставлять используемые на подготовительной фазе **1 данные измерений. При этом одна из них (исходная установка) имеет датчик, который отсутствует у другой (целевой установки). Исходная установка является источником используемых на подготовительном шаге **2 данных.

Относящиеся к областям значений различия между базовыми установками характеризуются разными режимами работы. Такие различия в работе обусловлены различными параметрами установок или различными условиями, влияющими на осуществляемый установкой базовый процесс.

Если продолжить пример с доменной печью, две одинаковые установки обрабатывают один и тот же металл, используют одни те же материалы (например, руду и кокс), имеют в сопоставимых местах датчики для одних и тех же параметров и тому подобное.

Но есть и отличия: первая печь может быть больше второй (разница в размерах) или печи могут иметь немного разную форму, что приводит к немного разным значениям измерений, например, объема расплавленного материала, времени плавления и тому подобному. Различие в размерах ярко выражено в имеющих разную высоту печах (см., например, фиг. 8).

Относящиеся к областям значений различия приводят к разным данным измерений для тех же эквивалентных параметров.

Кроме того, относящиеся к областям значений различия являются результатом применения различных способов измерения.

Вследствие относящихся к областям значений различий компьютер применяет преобразование данных.

Для наглядного примера кратко обратимся к музыке. Как органные трубы, так и доменные печи генерируют звук из потока воздуха. Более длинные органные трубы играют на более низких частотах, чем короткие. Но человек научился распознавать мелодии, то есть последовательности тонов с различными относительными частотами. Мелодии являются инвариантными к абсолютным частотам.

Поскольку в качестве данных измерений могут быть использованы издаваемые печью звуки, преобразование выделяет мелодию из высоты тона. Другими словами, на соответствующим образом приспособленный преобразователь можно подать два временных ряда 1{X1... XM} и 2{X1... XM} из первой и второй печей и получить мелодии (инвариантные к областям значений), которые могут быть подвергнуты дальнейшей обработке (например, сравнению).

Вид данных

Для простоты изложения в описании изначально предполагается, что данные измерений имеют одинаковый вид данных. Предполагается, что данные измерений во временных рядах состоят только из скаляров. Факультативное видовое приспосабливание будет рассмотрено в связи с фиг. 6.

Относимость к определению индикатора Z' параметра

Для упрощения объяснения в описании изначально предполагается, что все параметры от x_1 до x_N могут влиять на параметр z. Поэтому компьютер обрабатывает их все. Факультативно, соответствующие различия могут быть учтены, подробности пояснены в описании в связи с фиг. 7.

Шаги создания конфигурации

В последующем изложении описаны шаги, которые один или несколько компьютеров выполняют для получения конфигурации. Описание чертежей следует очередности шагов. Однако удобно начать обзор фиг. 2-4 с конца.

В ходе своей работы (см. фиг. 4) производственная установка 123 предоставляет данные измерений нейронной сети 373, которая предоставляет индикатор Z' параметра.

Сеть 373 обеспечивает возможность предоставления Z', поскольку:

- Конфигурация 283 была получена ранее в обучающей фазе (фиг. 3) путем обработки исторических данных измерений, которые являются практически полными (содержат данные для z), но происходят от производственной установки 112 (для которой имеются подходящие для z данные). Более подробно, временные ряды 212 и 272 включают в себя общие данные измерений 1{{X1... XM}}N и другие данные 1{Z1... ZM}}.
- Несмотря на неспособность имеющей конфигурацию 283 сети 373 к прямой обработке содержащих данные 2{{X1... XM}}N временных рядов 223 измерений,

преобразователь 323 преобразует содержащий данные 2{{X1... XM}}N временной ряд 223 в промежуточный временной ряд 243 2{{Y1... YM}}K. Преобразователь 323 может предоставлять промежуточные временные ряды 243, поскольку:

- Во время подготовительной фазы (см. фиг. 2) предшествующий преобразователю 323 преобразователь 311/321 сконфигурирован для преобразования временных рядов {{X1... XM}}N в промежуточные временные ряды {{Y1... YM}}K. YM}}K.
- Конфигурация преобразователя 311/321 выполнена с помощью дискриминатора 361, который оптимизирует общие правила преобразования так, что промежуточные временные ряды перестают быть специфичными для относящихся к областям значений различий между физически различными установками 111 и 121.

Обучение сети 372 для получения конфигурации 283 (см. фиг. 3) является возможным поскольку:

- Обучаемая сеть может получать исторические данные измерений (от аналогичной установки, в данном случае, от установки 112), поскольку преобразователь 322 обеспечивает возможность преобразования этих данных. Необходимость в преобразовании временных рядов 1{Z1... ZM} отсутствует.

Происхождение данных измерений

Для упрощения иллюстрации фиг. 2-4 показывают данные измерений как поступающие от датчиков. Однако некоторые данные могут поступать и через человеко-машинные интерфейсы или тому подобное. Для удобства датчики обозначены так же, как предоставляемые ими данные измерений: X1... XN, а также Z, символы размещены в маленьких скругленных квадратах.

Фиг. 2 показывает производственные установки и компьютерные модули в подготовительной к получению (421) набора правил преобразования фазе **1.

Производственные установки 111 и 121 имеют датчики X1-XN (общие датчики, поскольку они предоставляют данные об общих параметрах технологического процесса). Установка 111 имеет датчик Z (выделен полужирным шрифтом) для получения одномерных временных рядов 2111{{X1... XM}}N измерений, а установка 121 такого датчика не имеет.

Между аналогами может быть осуществлен «обмен» содержащей данные Z измерений информацией, но не напрямую, а в процессе установления правил.

Установка 111 является «исходной», а установка 121 является «целевой». Установка 111 является единственной имеющей датчик Z установкой.

Эквивалентность параметров

Параметр xn технологического процесса присутствует в обеих базовых установках (1xn для исходной и 2xn для целевой, см. матрицу на фиг. 1). Например, обе установки имеют параметр температуры в верхней части печи. Параметр x(n+1) технологического процесса, представленный, например, температурой в нижней части, также доступен на обеих установках.

Другими словами, присутствующие на обеих базовых установках параметры технологического процесса являются эквивалентными параметрами. Эквивалентные параметры обозначены одинаковым индексом n переменной.

Эквивалентность параметров приводит к эквивалентности данных измерений.

Датчик Xn установки 111 («исходной») и датчик Xn установки 121 («целевой») являются эквивалентными, поскольку получаемые данные измерений относятся к одному и тому же параметру, например, оба являются датчиками температуры в верхней части. Датчик Xn установки 111 и датчик X(n+1) установки 121 не являются эквивалентными.

Параметр является единственным (например, температура в верхней части), а данные измерений множественными (например, две печи).

Эквивалентность датчиков

Датчики разных установок, обеспечивающие измерение эквивалентных параметров, являются эквивалентными датчиками. Эквивалентные датчики не обязательно должны работать по одному и тому же принципу.

Например, как датчик Xn (исходной установки), так и Xn (целевой установки) предоставляют температуру верхнего слоя газа. При этом Xn (исходная установка) делает это инвазивным способом (путем перемещения балки с закрепленными на ней датчиками над доменной шихтой), а Xn (целевая установка) - с помощью звуковых излучателей и приемников над доменной шихтой (то есть, над загрузкой печи).

Исторические данные

В общем случае для обучения инструмента для машинного обучения (например, сети) требуются исторические данные. Работа преобразователя 311/321 начинается с приема

- многомерного временного ряда 211,1{{X1... XM}}N) измерений, который содержит исторические данные измерений базовой установки 111, и
- второго многомерного временного ряда 221, 2{{X1... XM}}N) измерений, который содержит исторические данные измерений от физически другой второй базовой установки 121.

Хотя на чертеже прием показан стрелками, специалист в данной области техники сохраняет временные ряды в базах данных. Отсутствует необходимость в дальнейших пояснениях в настоящем документе.

Однако только лишь загрузка {{X1... XM}}N в обучаемый инструмент (например, преобразователь для обучения правилам) является недостаточной для M*N элементов данных (см. матрицу на фиг. 1). Возможны два варианта:

Первый - исторические данные могут поступать от нескольких физических установок 111/112 (не только от двух, но и от многих других установок). Этот вариант показан посредством пунктирных повторений блоков на диаграмме. Однако с практической точки зрения (разнородный парк установок) такой подход для печей является неприменимым.

Второй - физические установки 111/112 предоставляют данные повторяющимся образом. Второй вариант является предпочтительным.

Фиг. 2 показывает множественные ряды {{...}}N (а также {{...}}K) посредством двух скругленных прямоугольников (в виде наложения). На чертеже также изображена стрелка Q. Другим обозначением является {{{...}}} Q.

Q - представляет собой число многомерных временных рядов измерений. Удобно иметь одинаковое Q для всех установок. Однако на практике это не всегда является возможным. Установка может оказаться «младше» (Q меньше), чем аналогичная установка (Q больше).

В соответствии с приведенным выше примером, при $\Delta t*M = 8$ часов установка за год собирает $Q = 3*365$ многомерных временных рядов измерений. Другими словами, Q может иметь порядок величины 1000 в год. При этом отсутствует необходимость в учете всех имеющихся данных. Иногда установка может находиться в режиме технического обслуживания или ремонта, и тогда исторические данные измерений оказываются недоступными.

Для наглядности удобно использовать 8-часовой временной интервал. Однако для вычисления признаков многомерного временного ряда компьютер

может применять оконный интервал T_WIND0W. Такой оконный интервал может быть определен как движущееся временное окно длительностью несколько часов (например, 8 часов). Оконный интервал может перемещаться по данным, собранным за более длительный период (например, за год) с перекрытием (T_OVERLAP). Удобными соотношениями между T_WIND0W и T_OVERLAP являются, например, T_WIND0W > T_OVERLAP (или даже равны). Удобно, когда T_WIND0W составляет порядка 6-10 часов, а T_OVERLAP - от 1 до 3 часов.

Удаление относящихся к областям значений различий из исторических данных

Несмотря на отмеченную эквивалентность (по параметрам, датчикам), исторические данные все же отражают относящиеся к областям значений различия.

Компьютер получает (шаг 421 на фиг. 5) набор правил 251 преобразования (на фиг. 2) /252 (на фиг. 4) путем обработки многомерных временных рядов 211 и 221 измерений. Правила 251/252 преобразования позволяют преобразователю 311/321 преобразовывать соответственно многомерные временные ряды 211, 221 ({{...}}) измерений в многомерные временные ряды 231 и 241 признаков. Эти временные ряды 1{{Y1... YM}}K и 2{{Y1... YM}} признаков являются инвариантными к относящимся к областям значений различиям базовых установок 111/121.

Другими словами, извлечению подвергают инвариантные к областям значений признаки.

Использование множественных многомерных временных рядов измерений (см. Q) только лишь упомянуто, преобразование их не изменяет.

Преобразователь 311/321 изображен в данном случае посредством двух блоков, поскольку временной ряд 211 преобразуют во временной ряд 231, а временной ряд 221 преобразуют во временной ряд 241. При этом не имеет значения, выполняют ли преобразование параллельно (как на чертеже) или последовательно. Оба преобразователя используют общие правила 251/252.

Правила получают путем обучения. В данном случае отсутствует необходимость в привлечении эксперта-человека. Другими словами, приспосабливание к области значений (при наличии признаков вместо данных измерений) происходит без участия эксперта. Использование экспертных

примечаний или меток не требуется. Факультативно, эксперт может задавать некоторые правила вручную, на основе имеющихся экспертных знаний. Задание некоторых правил вручную может дополнительно способствовать обеспечению сходимости обучения.

Правила преобразования

Поскольку сеть подвергают обучению на обучающих данных из различных областей значений, имеет место перенос обучения.

Упрощенно, имеется преобразование, которое соответственно преобразует первый и второй многомерные временные ряды 211, 221 измерений в первый и второй многомерные временные ряды 231, 241 признаков. Многомерные временные ряды 231, 241 признаков являются инвариантными к относящимся к областям значений различиям базовых установок 111, 121.

Другими словами, в результате преобразования извлекают признаки, которые уже не являются специфическими для данной области значений. Временные ряды измерений можно рассматривать как принадлежащие первому пространству (или входному пространству), а временные ряды признаков - как принадлежащие второму пространству (или пространству признаков). Входное пространство является чувствительным к относящимся к областям значений различиям, а пространство признаков - нет.

Можно предположить наличие причинно-следственной связи между пространством X и пространством Y, которое представляет собой непрерывное пространство для регрессии.

Маргинальные распределения в обеих областях значений (исходный=s, целевой=t) являются подобными: $ps(ys|T(xs)) \approx pt(yt|T(xt))$, где ys - выходные данные в исходной области значений («s»), а yt - выходные данные в целевой области значений («t»).

Правила преобразования подчиняются принципу Y = T(X). Более подробно, преобразователи 311, 321 работают в соответствии с временными интервалами Δt: преобразователь 311 преобразует вектор 1{X1}N данных в вектор 1{Y1}K данных, а преобразователь 321 преобразует вектор 2{X2}N данных в вектор 2{Y2}K данных для t=1. Далее они преобразуют 1{X2}N в 1{Y2}K и 2{X2}N в 2{Y2}K для t=2, и так далее до t=M.

Поскольку переменные {{X1... XM}} N не обязательно совпадают с переменными {{Y1... YM}}K, преобразование может изменять кардинальное число переменных (обычно N > K).

Хотя правила преобразования представлены в данном случае для операций с данными в момент времени tm (по вертикали - обработка векторов данных для временных интервалов tm), правила могут быть расширены для обработки данных из предшествующих временных точек (m-1), (m-2) и тому подобное. Специалистам известны такие рекуррентные сети.

Поскольку преобразование уже описано в данной области техники, специалист может выбрать одну из существующих схем преобразования.

Для удобства следует упомянуть следующие дополнительные публикации:

- Hochreiter et al, «Long Short-Term Memory», in Neural ComputationNovember 1997, https://doi.Org/10.1162/neco.1997.9.8.1735 (1997)
- Ashish Vaswani et al, «Attention Is All You Need», Advances in Neural Information Processing Systems 30, NIPS 2017, (2017)
- Anastasia Borovykh et al, «Conditional Time Series Forecasting with Convolutional Neural Networks», arXiv:1703.04691 (2017)
- Aaron van den Oord, et al., «WaveNet: A generative model for raw audio», arXiv:1609.03499, (2016).

Признаки для мультимодальных многомерных временных рядов извлекают с помощью алгоритма, такого как, в том числе, но им не ограничиваясь, рекуррентная нейронная сеть (Хохрейтер), потенциально совмещенная со сверточными нейронными сетями, или преобразователями (Васвани), или WaveNet (нейронной сетью для генерации необработанного звука) (Боровых, ван ден Оорд).

Функция T преобразования может быть определена путем минимизации несходства между обоими условными по переменной класса распределениями $ps(t(xs)|ys) \approx pt(t(xt)|yt)$.

Функцию t() преобразования для определения инвариантных к областям значений признаков обычно находят путем обучения состязательной модели глубокого обучения, где преобразование сходится для генерации признаков, в которых исходная или целевая области значений не могут быть отличимы друг от друга, а потери предсказывающей выходной результат регрессионной модели минимизированы.

Более подробная информация представлена в следующей публикации: Y. Ganin, et al., «Domain-adversarial training of neural networks», Journal of Machine Learning Research, vol. 17, no. 59, pp. 1-35, (2016).

Пример обучения преобразователя

В описании предполагается, что преобразователь уже получил правила 251/252. Далее описан факультативный подход к получению правил путем применения состязательного (глубокого) обучения.

Такой подход может быть сведен к нескольким шагам способа, которые представлены как подшаги шага 421 получения правил (см. фиг. 5). Получение 421 набора правил 251/252 преобразования может быть осуществлено посредством компьютера, который итеративно выполняет следующие действия:

На шаге 431 преобразования, и с использованием набора правил предварительного преобразования, компьютер преобразует (первый) многомерный временной ряд 211 измерений в (первый) многомерный временной ряд 231 признаков, а также преобразует (второй) многомерный временной ряд 221 измерений во (второй) многомерный временной ряд 241 признаков. Подробности были описаны выше.

На шаге 441 различения компьютер с помощью дискриминирующего модуля 361 проводит различение (или «дискриминирует») происхождение первого и второго временных рядов 231, 241 признаков как происходящих из (первого) и (второго) многомерных временных рядов 211, 221 измерений. На первых итерациях дискриминатор 361 может «легко» их различать. На фиг. 2 этот процесс представлен в виде схемы с указанием времени обучения (общего времени проведения итераций). Слева (раннее обучение) дискриминатор способен к различению: преобразования X в Y из 1 (первая установка/область значений) или из 2 (вторая установка/область значений). Справа дискриминатор не способен к различению.

На шаге 451 изменения компьютер изменяет набор правил предварительного преобразования (или изменяет правила текущей итерации).

Компьютер итеративно выполняет шаги 431, 441 и 451 до тех пор, когда дискриминатор 361 перестает различать источник данных (то есть, 1 или 2). Для определения его неспособности к различению источника данных, дискриминатор 361 может вычислять потери ADLOSS. Дискриминатор 361 продолжает работать

(и позволяет изменять правила) до тех пор, когда ADLOSS достигает максимального значения.

Правила 251/252 преобразования становятся окончательными правилами, которые подлежат применению на последующих шагах.

В данном конкретном подходе правила получают в процессе обучения с такой целью обучения, которая может представляться парадоксальной.

Другими словами, комбинацию преобразователя 311/321 и дискриминатора 361 обучают (с помощью изменения настроек для правил) подразделению (или «предсказанию») принадлежности признаков к исходной установке или к целевой установке.

Другими словами, одно и то же устройство кодирования (или применяющий подготовленные в ходе итераций правила преобразователь) используют совместно для исходной и целевой областей значений (или для первой и второй базовой установки), причем предполагается, что устройство кодирования должно извлекать инвариантные к областям значений представления с помощью состязательного дискриминатора (то есть, дискриминатора 361).

Для получения более подробной информации можно ознакомиться со следующими публикациями.

- Zhang, et al., «Domain-Invariant Adversarial Learning for Unsupervised Domain Adaption», arXiv:1811.12751, (2018)
- Xie, et al., «Learning Semantic Representations for Unsupervised Domain Adaptation®», Proceedings of the 35th International Conference on Machine Learning, PMLR 80:5423-5432, (2018)
- Avisek Lahiri, et al., «Unsupervised Domain Adaptation for Learning Eye Gaze from a Million Synthetic Images: An Adversarial Approach», In Proceedings of the 11th Indian Conference on Computer Vision, Graphics and Image Processing (2018)

Состязательное обучение также рассмотрено в работе Goodfellow, Ian J., Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron C., and Bengio, Yoshua. Generative adversarial nets. NIPS, 2014.

Получение правил с помощью обучающего подхода относится к подготовительной фазе **1. Это не является тем же процессом, что обучение сети, о котором будет рассказано далее.

Фиг. 3 показывает базовую производственную установку 112, преобразовательный модуль 322 (сокращенно «преобразователь 322») и сеть 372 (в процессе обучения) в обучающей фазе **2.

Преобразователь 322 преобразует (шаг 452 на фиг. 5) многомерные временные ряды 212 измерений в многомерные временные ряды 323 признаков. Временной ряд 212 может быть таким же, как и временной ряд 211 (см. фиг. 2). Преобразователь 322 применяет правила 252 преобразования (см. правила 251 на фиг. 2, когда они доработаны). Как показано посредством нескольких наложенных прямоугольников, преобразователь 322 обрабатывает несколько многомерных временных рядов измерений (см. представленное на фиг. 2 Q).

Компьютер принимает (шаг 462 на фиг. 5) одномерный временной ряд 272 измерений (то есть, 1{Z1... ZM}), который содержит данные (Z) измерений другого параметра (z) технологического процесса от базовой установки 112, обычно также в виде нескольких рядов. В этом смысле производственная установка 112 выступает в роли базовой установки, предоставляющей данные реальных измерений (например, от датчиков X1-XM и от датчика Z).

Сеть 372 обучают (шаг 472 на фиг. 5) с использованием первого многомерного временного ряда (232) признаков на входе и с использованием одномерного временного ряда (272) измерений на выходе.

Специалисты в данной области техники могут проводить обучение с использованием соответствующих гиперпараметров, целей оптимизации тому подобного. Например, обучение завершается, когда функция потерь достигает определенного заранее заданного значения. В процессе обучения промежуточное расчетное значение LOSS (или «сетевые потери») изменяется, и в конце концов достигнет минимума, так что веса и значения смещения могут быть перенесены в конфигурацию. Другими словами, LOSS является управляющим параметром для циклов внутрисетевой обработки.

LOSS можно также использовать для управления внешними по отношению к сети 372 циклами. На чертеже показано, что промежуточное расчетное значение LOSS может быть предоставлено посредством сети 372. Как будет пояснено в связи с фиг. 7, LOSS можно использовать посредством обучателя 380 селектора в другом цикле. Другими словами, LOSS также может быть управляющим параметром для одного или нескольких связанных с сетью циклов обработки.

Кроме того, LOSS можно предоставлять преобразователю (см. фиг. 2) в качестве оптимизирующего правила преобразования параметра.

Фиг. 4 показывает производственную установку (на фиг. 1) и преобразовательный модуль 323 (см. фиг. 3) во время рабочей фазы **3 для предоставления индикатора Z' параметра. Фиг. 4 представляет собой обновленный вариант фиг. 1, показанные на которой подробности были описаны выше.

Компьютер принимает (см. шаг 513 на фиг. 5) многомерные временные ряды 223 измерений (также обозначенные как 2{{X1...XM}}N), которые содержат данные измерений для производственной установки 123. В данном случае установка 123 является установкой под наблюдением, которая не имеет датчика для измерения параметра z.

Преобразователь 323 преобразует многомерный временной ряд 223 измерений в многомерный временной ряд 243 признаков (также обозначенный как 2{{Y1...YM}}K).

Сеть получает многомерные временные ряды 243 признаков и выдает индикатор Z' параметра. Сеть 373 показана в конфигурации 283.

Следует отметить, что в отличие от фиг. 2-4, временные ряды являются отдельными временными рядами (то есть, для конкретной рабочей смены и тому подобного).

На фиг. 5 показаны блок-схемы компьютерно-реализуемого способа 400 обучения нейронной сети 372/373 и компьютерно-реализуемого способа 500 использования сети в процессе работы производственной установки 123.

Способ 400 представляет собой компьютерно-реализуемый способ обучения нейронной сети 373. Сеть 373 обучают для последующей обработки многомерных временных рядов измерений (см. 223 на фиг. 1), которые содержат данные измерений, представляющие отдельные параметры (см. 183 на фиг. 1) технологического процесса производственной установки (см. 123 на фиг. 1) и, таким образом, для получения индикатора Z' параметра для другого параметра (см. 173, z на фиг. 1) технологического процесса производственной установки.

На шаге 411 приема компьютер принимает первый многомерный временной ряд измерений, который содержит исторические данные измерений от первой базовой установки, и второй многомерный временной ряд измерений, который

содержит исторические данные измерений от физически другой второй базовой установки (1{{X1...XM}}N) и 2{{X1... XM}}N), см. левую часть фиг. 2).

На шаге 421 получения компьютер путем обработки первого и второго многомерных временных рядов измерений получает набор правил 251/252 преобразования, которые правила 251/252 преобразования позволяют преобразователю (см. 311/321 на фиг. 2) преобразовывать соответственно первый и второй многомерные временные ряды измерений в первый и второй многомерные временные ряды признаков (см. 231, 241 на фиг. 2, также обозначенные как 1{{Y1... YM}}K, 2{{Y1... YM}}K). Многомерные временные ряды признаков являются инвариантными к относящимся к областям значений различиям первой и второй базовых установок (см. 111 121, фиг. 2). (Обе базовые установки принадлежат к различным областям значений, пояснения подробностей приведены выше).

На шаге 452 преобразования компьютер преобразует первый многомерный временной ряд измерений в первый многомерный временной ряд признаков с помощью преобразователя (322, на фиг. 3), который применяет правила преобразования.

На шаге (462) приема компьютер принимает от первой базовой установки (см. 112 на фиг. 3) одномерный временной ряд измерений, который содержит данные Z измерений другого параметра z технологического процесса.

На шаге 472 обучения нейронной сети 372 компьютер обучает сеть, используя на входе сети первый многомерный временной ряд признаков, а на выходе сети - одномерный временной ряд измерений.

Фиг. 5 также показывает дополнительные детали (пунктирные блоки) для получения 421 набора правил преобразования. Компьютер итеративно выполняет шаги 431, 441 и 451.

На шаге 431 преобразования компьютер преобразует первый многомерный временной ряд измерений в первый многомерный временной ряд признаков и преобразует второй многомерный временной ряд измерений во второй многомерный временной ряд признаков с использованием набора правил преобразования, которые являются предварительными правилами.

На шаге 441 различения компьютер запускает дискриминатор 361 для различения (или дискриминации) происхождения первого и второго временных

рядов признаков как первого и второго многомерных временных рядов измерений.

На шаге 451 изменения компьютер изменяет набор предварительных правил преобразования.

Итеративное выполнение шагов останавливают, когда дискриминатор (361) более не может различать источник данных (то есть, цикл REPEAT UNTIL или WHILE).

Подробности шагов способа 400 объяснены в контексте описания.

Относительно компьютерно-реализуемого способа 500 использования сети во время работы производственной установки, на чертеже показаны шаги и дана отсылка на фиг. 4.

На шаге 513 приема компьютер принимает многомерный временной ряд измерений, который содержит данные измерений для наблюдаемой производственной установки, на шаге 523 преобразования преобразует многомерный временной ряд измерений в многомерный временной ряд признаков, а на шаге 533 работы использует сеть 373 для получения индикатора Z' параметра.

Следует еще раз отметить, что компьютеры для выполнения способов 400 и 500 могут быть физически разными. Общие данные представлены конфигурацией 283 (полученной в результате обучения).

Повышение точности

Как уже было указано выше, точность индикатора Z' соответствует различию с данными Z измерений (если они имеются). Упрощенно, меньшее различие означает более высокую точность.

В последующем описании рассмотрены два подхода, которые позволяют повысить точность. Один подход (приспосабливание) увеличивает число обрабатываемых переменных, а другой (селекция) уменьшает их число.

На первый взгляд, два подхода противоречат друг другу, однако они являются взаимосвязанными. Для удобства в описании они представлены по отдельности, а их взаимосвязь рассмотрена в дальнейшем описании.

Приспосабливание вида данных

До сих пор в описании рассматривался подход для данных измерений во временных рядах, состоящих только из скаляров, где {X1... XM} является последовательностью из M последовательных скалярных значений. При этом не

имеет значения, представляет ли компьютер скаляр в виде целого или вещественного числа.

Теперь в описании рассматривается, как могут быть приспособлены для обработки данные измерений других видов.

Фиг. 6 показывает факультативный модуль адаптера 393 вида (или сокращенно «адаптер»), который факультативно реализует видовое приспосабливание данных измерений. В качестве примера на чертеже показан адаптер 393 в рабочей фазе **3, но адаптер 393 может работать и в других фазах.

Адаптер 393 принимает многомерные временные ряды 213 измерений (также обозначенные как {{X1...XM}}) от производственной установки (в качестве примера показана установка 113, но происхождение данных «1» или «2» не имеет значения). Термин «принимает» соответствует шагу 411 способа, равно как и шагу 513 способа (см. фиг. 5). Соответственно, адаптер 393 может быть реализован как часть выполняющего прием модуля.

Многомерный временной ряд {{X1... XM}}N измерений включает в себя одномерные временные ряды {X1... XM} измерений, которые представляют собой последовательности скалярных или нескалярных элементов. В примере одномерные временные ряды {X1... XM}n должны состоять из нескалярных элементов Xn.

В качестве примечания следует отметить, что во многих ситуациях 1{Z1... ZM} является исключительно скалярным. Однако это не является обязательным, так что 1{Z1... ZM} может быть представлен также вектором или тому подобным.

Адаптер 393 передает содержащие скаляры временные ряды другим модулям, по существу, неизменными. Эти временные ряды даже не должны проходить через адаптер.

Адаптер 393 обрабатывает нескалярные элементы Xn индивидуально, но по правилам для каждого n. На чертеже нескалярный элемент Xn изображен очень условно. Таким образом, компьютер может обрабатывать элементы и присваивать им скалярное значение (посредством классификации), которое здесь представлено как Xn~. Знак ~ просто указывает на то, что присвоение произошло.

В первом варианте нескалярные элементы представляют собой образы (то есть, матрицы с обозначающими цвета пикселями). Например, образы показывают смотровое отверстие установки, сфотографированное через каждый интервал Δt времени (временем экспозиции образа можно пренебречь).

В примере с доменной печью при осмотре используют фурмы, которые позволяют наблюдателю увидеть огонь внутри печи. Огонь регулярно возникает в процессе эксплуатации. Упрощенно, изображение с огнем можно закодировать как 1, а изображение без огня - как 0. Полученный временной ряд может иметь вид $\{...\}\sim = \{0, 0,1,1,1, 0 ... 0, 0\}\sim$, и компьютер продолжает обработку (посредством последующих шагов, таких как получение 421 правил, преобразование 523, см. фиг. 5, или также прием 513).

Во втором сценарии нескалярными элементами являются звуковые последовательности. Например, Xn может быть представлен звуковой записью от микрофонного датчика с длительностью временного интервала Δt = 15 мин. Специалист в данной области техники может применять соответствующую обработку звука, например, дискретизировать его с частотой 20 кГц, что дает 15*60*20 000 аудиозамеров в секунду.

Затем адаптер 393 перерабатывает эти миллионы замеров в один скаляр. Например, скаляр может указывать на: повышение тона во время Δt, снижение тона во время Δt, постоянство тона во время Δt, повышение и снижение тона во время Δt (как в примере) и тому подобное.

Адаптер 393 может быть реализован посредством нейронной сети, которая была обучена ранее, возможно, под наблюдением (человека-эксперта). Например, адаптер 393 может быть обучен на содержащих изображения огня (в доменных печах) изображениях с экспертными примечаниями, на последовательностях звуков (из доменных печей) с экспертными примечаниями и тому подобном. Адаптер 393 в целях идентификации абстрактного представления образа также может быть обучен и неконтролируемым образом.

За счет этого может быть обеспечена максимизация информационного содержимого образа. Последовательность образов в результате этого процесса может быть сведена к нескольким последовательностям скаляров.

Присвоение скаляра не обязательно означает, что данные измерений связаны с индикатором Z'. На чертежах показано, что для приспосабливания могут быть использованы по-разному обученные сети, поэтому скалярные

значения могут быть различными Xn~ и Xn~~ для одного и того же входного сигнала Xn. Цифры «6» и «7» на чертежах приведены только в виде примера. Последствия использования разных скаляров для одного и того же входа рассмотрены ниже.

Выбор относимости

Некоторые параметры процесса не связаны с параметром z. Другими словами, общий набор параметров (от x_1 до x_N, имеющий кардинальное число N переменных) технологического процесса может быть разделен на

- первый поднабор H параметров x_1 - x_H технологического процесса, связанных с параметром z («набор относящихся к делу параметров»), и
- второй поднабор (N-H) параметров технологического процесса, не связанных с параметром z.

Если воспользоваться музыкальным примером, то в дальнейшем экскурсе не имеет значения, играют ли мелодию медленно или быстро, по меньшей мере для слушателя, мелодия остается той же.

В идеальной ситуации сеть 373 должна быть обучена попросту игнорировать данные измерений для параметров второго поднабора. В реальных сценариях обучения некоторые данные измерений (второго поднабора) могут по-прежнему вносить вклад в индикатор Z'.

Следует отметить, что обучение может быть проведено без участия людей-экспертов. Поэтому отсутствует необходимость введения в обучение такого экспертного контроля.

Эксперты по производственным установкам могут определять соответствующий набор параметров на основе опыта. В компьютеры поступают только данные измерений, относимость которых определена экспертами. Привлечение эксперта на этом шаге является однократным.

Например, потребляемая печью мощность может не влиять на температуру (типовой параметр z), а данные о потреблении можно не учитывать (посредством отсутствия обработки измерений потребления).

В дополнение и в качестве альтернативы к ручному предварительному выбору, определение относимости может быть выполнено посредством компьютера путем выбора данных измерений. Пример разъяснен ниже.

Фиг. 7 показывает факультативный модуль 383 селекции данных (или сокращенно «селектор»), который факультативно реализует выбор относящихся

к делу данных измерений. В верхней части чертежа показан селектор 383 в рабочей фазе **3, а также показана его настройка в фазах **1 и **2 путем итеративного выполнения некоторых шагов способа на фиг. 5.

В процессе работы селектор 383 принимает многомерный временной ряд 223 измерений (см. фиг. 1 и 3, также обозначенный как {{X1... XM}}N) и направляет измененный многомерный временной ряд 223 измерений (также обозначенный как {{X1... XM}}H) на преобразователь 323 (см. фиг. 4).

Изменение представляет собой выбор (то есть, положительный выбор из N переменных H переменных в {{...}}H, причем H < N). Другими словами, это соответствует выбору относящегося к делу набора параметров (но на уровне данных измерений). На чертеже выбор проиллюстрирован посредством показа меньшего числа символов датчика на выходе селектора 383 (по сравнению с установкой).

Выполнение способа 500 (см. фиг. 5) с уменьшенным набором данных измерений способствует повышению точности предоставления индикатора Z'. Такой подход может показаться парадоксальным: селектор блокирует часть данных для обработки сетью, но при этом точность может быть повышена. Однако при выборе отсеиваются данные измерений, которые сеть 373 потенциально не может игнорировать и которые потенциально могут снизить точность. Еще раз следует отметить неконтролируемый характер обучения сети (см. фиг. 2-4).

Селектор 383 может быть настроен с помощью процесса обучения, который использует функцию потерь сети 372/373 в качестве значения для минимизации. Такой подход символически представлен посредством модуля 380 обучения селектора (или сокращенно «обучателя селектора»).

Обучатель 380 селектора взаимодействует с другими модулями, например, с сетью 372 (которая проходит обучение и принимает LOSS) и выполняет представленный блок-схемой способ.

В ходе выполнения итераций до тех пор, когда промежуточное значение LOSS (другими словами, функция потерь) принимает минимальное значение («min loss», YES), обучатель 380 селектора изменяет (предварительный) выбор ({{...}}H ∈ {{...}}N) и позволяет компонентам выполнять способ 400 (см. фиг. 4), то есть (факультативно) принимать 411, получать 421 правила, преобразовывать

452, (факультативно) принимать 462 и обучать 472. Прием является факультативным, поскольку принятые данные остаются неизменными. Другими словами, предварительный выбор {{...}}H переменных заменяют другим выбором до тех пор, когда LOSS принимает минимальное значение для окончательного выбора.

Объяснение является упрощенным, и специалист может реализовать способ соответствующим образом. Например, отсутствует необходимость в повторении шага 462 приема (поскольку 1{Z1...ZM} не подлежит выбору).

Как только функция потерь достигает минимума, описанные на фиг. 1-5 операции подлежат выполнению с использованием {{...}}H вместо {{...}}N (то есть, в ходе рабочей фазы **3, как объясняется для верхней части чертежа).

Факультативно, работа селектора 383 может быть улучшена за счет выбора некоторых переменных временного ряда измерений и их объединения или сведения в результирующий временной ряд измерений. Например, селектор 383 может выбрать {...}1, {...}2 и {...}3 (например, температуры из трех мест) и объединить их в новый одномерный временной ряд измерений (который является частью {{...}}H).

Специалист в данной области техники может реализовать такие выбор и объединение с помощью анализа главных компонент.

Рассмотрение

Модули, выполняющие приспосабливание вида данных (см. фиг. 6) и выбор относимости (см. фиг. 7), в фазе **3 работают независимо друг от друга.

Однако для экономии вычислительных ресурсов удобным является наделение селектора 383 данных приоритетом в работе над адаптером 393 вида, по меньшей мере, в рабочей фазе **3. Например, отсутствует нужда в переработке звуковой записи в скаляр (см. фиг. 6), а затем в фильтровании скаляра посредством селектора 383.

Хотя приспосабливание путем переработки в скалярный вид позволяет сети обрабатывать больше данных, при некоторых приспосабливаниях могут возникать ошибки. Посредством присваивания скалярных значений, адаптер обеспечивает возможность применения различным образом обученных сетей. Одни результаты могут оказаться относящимися к Z', другие - нет. Например, анализ звуковых последовательностей с разделением их на такие категории, как повышение/снижение/постоянство тона может не иметь отношения к Z', а анализ

с разделением на другие категории, такие как осциллирующий/стабильный звук может быть действительно связан с Z'. В этом смысле, полученные различными способами скаляры Xn~ и Xn~~ (см. фиг. 6) можно рассматривать как выбор относимости (фиг. 7).

Сетевые потери

Описание теперь ненадолго возвращается к фиг. 2-3. Как уже было объяснено, правила преобразования получают в подготовительной фазе **1 так, что преобразователь предоставляет временные ряды признаков, которые являются инвариантными к относящимся к областям значений различиям. В качестве критерия выступают состязательные потери (ADLOSS, см. фиг. 2).

Для дальнейшего повышения точности правила преобразования могут быть расширены за счет учета сетевых потерь (LOSS, см. фиг. 3). Таким образом, компьютер может работать в цикле, состоящем из подготовительной фазы **1 и обучающей фазы **2 (которая выполняется как предварительное обучение).

Другими словами, итеративному выполнению подлежат шаги получения 421 набора правил преобразования и обучения 472 нейронной сети 372. Затем преобразователь 311/321 принимает предоставляемые обучаемой сетью сетевые потери (LOSS). Выполнение итераций может быть прекращено при достижении сетевыми потерями LOSS минимального уровня.

Для иллюстрации этого факультативного подхода, на фиг. 2 пунктирной линией показан прием LOSS. В то время как ADLOSS подлежат максимизации, LOSS - минимизации. Специалист может применять и другие методики.

Пример использования

На фиг. 8 показан пример использования производственных установок 111, 121 и 123, которые представляют собой доменные печи. Имеется несколько температурных параметров (обозначенных различными значениями от 200°C до 2000°C), а также других параметров технологического процесса (таких как значения давления, химический состав материалов и тому подобное). Относящиеся к областям значений различия проиллюстрированы на двух примерах. Различие в размерах показано посредством того, что печь 111 меньше печей 121 и 123. В печах 111 и 121 применены различные способы измерения температуры расплавленного материала (температуры горячего металла): дистанционный датчик 151 (символ камеры) в печи 111 позволяет получать временной ряд с замерами, доступными с временным интервалом Δt (например,

15 мин). Ручное измерение оператором печи 121 приводит к получению значений температуры через большие временные интервалы (например, 90 мин), о чем уже было упомянуто в посвященном уровню техники разделе.

В этом примерном варианте использования виртуальный датчик 153 получает индикатор Z' (для температуры) с временным интервалом Δt. Хотя этот виртуальный датчик проиллюстрирован маленьким и пунктирным символом камеры, он обозначает компьютер, выполняющий способ 500 (см. фиг. 5). Другими словами, на чертеже в общем показано использование компьютера, выполняющего (или осуществляющего) способ 500 (и обученного в соответствии с способом 400), для получения индикатора Z' параметра, представляющего собой параметр производственной установки. Такое использование выполняющего способ 500 компьютера можно рассматривать как использование виртуального датчика.

Печи 121 и 123 физически могут быть одинаковыми, но печь 123 работает в условиях уже после прохождения одной сетью 372/373 обучения. На чертеже не показан берущий замеры оператор печи 123, и тем самым показано, что данный вариант использования позволяет минимизировать участие человека в измерениях, и оператор может лучше сосредоточиться на работе печи. Другими словами, печь 121 может быть усовершенствована до печи 123 с помощью дополнительного виртуального датчика.

Использование виртуальных датчиков дает возможность сопоставления существующих производственных установок посредством измерения параметров технологического процесса, по которым в противном случае данные отсутствуют.

Использование искусственных данных измерений в качестве обучающих данных

Индикатор Z' может быть использован в качестве обучающих данных, в качестве дополнительных исторических данных.

Типовой компьютер

Фиг. 9 показывает пример типового вычислительного устройства, которое может быть использовано в рамках описанных в настоящем документе способов. Фиг. 9 представляет собой схему, на которой показан пример типового вычислительного устройства 900 и типового мобильного вычислительного устройства 950, которые могут быть использованы в рамках описанных в

настоящем документе способов. Вычислительное устройство 900 выступает в качестве примера различных видов цифровых компьютеров, таких как ноутбуки, настольные компьютеры, рабочие станции, карманные персональные компьютеры, серверы, сверхкомпактные серверы, мэйнфреймы и другие подходящие компьютеры. Типовое вычислительное устройство 900 может соответствовать компьютерной системе 100 на фиг. 1. Вычислительное устройство 950 может выступать в качестве примера различных видов мобильных устройств, таких как карманные персональные компьютеры, сотовые телефоны, смартфоны, системы помощи водителю или бортовые компьютеры автомобилей (например, автомобилей 401, 402, 403, см. фиг. 1) и другие подобные вычислительные устройства. Например, вычислительное устройство 950 может быть использовано пользователем (например, оператором доменной печи) в качестве фронтальной части для взаимодействия с вычислительным устройством 900. Показанные здесь компоненты, их соединения и взаимосвязи, а также их функции приведены только в качестве примера и не ограничивают варианты осуществления описанных и/или заявленных в настоящем документе изобретений.

Вычислительное устройство 900 включает в себя процессор 902, память 904, устройство 906 хранения данных, высокоскоростной интерфейс 908, соединяющий память 904 с высокоскоростными портами 910 расширения, и низкоскоростной интерфейс 912, соединяющий низкоскоростную шину 914 и устройство 906 хранения данных. Все компоненты 902, 904, 906, 908, 910 и 912 соединены между собой различными шинами и могут быть установлены на общей материнской плате или другим способом. Процессор 902 может обрабатывать подлежащие выполнению в вычислительном устройстве 900 команды, включая сюда сохраняемые в памяти 904 или на устройстве 906 хранения данных команды для отображения графической информации графического интерфейса пользователя на внешнем устройстве ввода/вывода, таком как подключенный к высокоскоростному интерфейсу 908 дисплей 916. В других вариантах осуществления могут быть использованы несколько процессоров и/или несколько шин, а также несколько видов и типов памяти. Кроме того, может быть подключено несколько вычислительных устройств 900, каждое из которых обеспечивает выполнение части необходимых операций

(например, в виде банка серверов, группы сверхкомпактных серверов или многопроцессорной системы).

Память 904 сохраняет информацию в вычислительном устройстве 900. В одном варианте осуществления память 904 представляет собой энергозависимый блок или блоки памяти. В другом варианте осуществления память 904 представляет собой энергонезависимый блок или блоки памяти. Память 904 может также представлять собой другую форму считываемого компьютером носителя, например, магнитный или оптический диск.

Устройство 906 хранения данных способно к обеспечению массового хранения данных для вычислительного устройства 900. В одном из вариантов осуществления устройство 906 хранения данных может иметь в своем составе машиночитаемый носитель или может быть представлено машиночитаемым носителем, таким как накопитель на гибком диске, накопитель на жестком диске, оптический дисковый накопитель или ленточный накопитель, флэш-память или другое подобное твердотельное устройство хранения данных, или массив устройств, включая сюда устройства в сети хранения данных или другие конфигурации. Компьютерный программный продукт может быть материально воплощен на носителе информации. Компьютерный программный продукт может также содержать команды, которые при их выполнении осуществляют один или несколько описанных выше способов. Носителем информации является считываемый компьютером или машиночитаемый носитель, например, память 904, устройство 906 хранения данных или память процессора 902.

Высокоскоростной контроллер 908 управляет операциями, требующими большой пропускной способности вычислительного устройства 900, а низкоскоростной контроллер 912 - операциями, требующими меньшей пропускной способности. Такое распределение функций является лишь примерным. В одном из вариантов осуществления высокоскоростной контроллер 908 соединен с памятью 904, дисплеем 916 (например, через графический процессор или ускоритель) и высокоскоростными портами 910 расширения, которые могут принимать различные платы расширения (не показаны). В данном варианте осуществления низкоскоростной контроллер 912 соединен с устройством 906 хранения данных и низкоскоростным портом 914 расширения. Низкоскоростной порт расширения, который может включать в себя различные коммуникационные порты (например, USB, Bluetooth, Ethernet, беспроводной

Ethernet), может быть соединен с одним или несколькими устройствами ввода/вывода, такими как клавиатура, указывающее устройство, сканер, или с сетевым устройством, таким как коммутатор или маршрутизатор, например, через сетевой адаптер.

Вычислительное устройство 900 может быть реализовано в различных формах, как показано на чертеже. Например, оно может быть реализовано в виде стандартного сервера 920 или нескольких таких серверов в группе. Оно также может быть реализовано как часть стоечной серверной системы 924. Кроме того, оно может быть реализовано в виде персонального компьютера, например, ноутбука 922. Альтернативно, компоненты вычислительного устройства 900 могут быть объединены с другими компонентами в мобильном устройстве (не показано), например, в устройстве 950. Каждое из таких устройств может содержать одно или несколько вычислительных устройств 900, 950, а вся система может состоять из нескольких взаимодействующих друг с другом вычислительных устройств 900, 950.

Вычислительное устройство 950 включает в себя процессор 952, память 964, устройство ввода-вывода, например дисплей 954, интерфейс 966 связи, приемопередатчик 968 и другие компоненты. Устройство 950 для предоставления дополнительной памяти также может быть снабжено устройством хранения данных, например микродиском или другим устройством. Все компоненты 950, 952, 964, 954, 966 и 968 соединены между собой с помощью различных шин, а несколько компонентов могут быть установлены на общей материнской плате или другим способом.

Процессор 952 может выполнять команды в вычислительном устройстве 950, включая сюда сохраняемые в памяти 964 команды. Процессор может быть реализован в виде набора микросхем, включающего в себя отдельные и множественные аналоговые и цифровые процессоры. Процессор может обеспечивать, например, координацию работы других компонентов устройства 950, такую как управление пользовательскими интерфейсами, выполняемыми посредством устройства 950 приложениями, а также беспроводной связью устройства 950.

Процессор 952 может взаимодействовать с пользователем через интерфейс 958 управления и соединенный с дисплеем 954 интерфейс 956 отображения. Дисплей 954 может представлять собой, например, TFT LCD (тонкопленочный

транзисторный жидкокристаллический дисплей) или OLED (органический светоизлучающий диодный) дисплей, или использующий другую подходящую технологию отображения дисплей. Интерфейс 956 дисплея для отображения графической и другой информации пользователю может включать в себя соответствующую схему управления дисплеем 954. Интерфейс 958 управления может принимать команды от пользователя и преобразовывать их для передачи в процессор 952. Кроме того, для обеспечения связи устройства 950 с другими устройствами в ближней зоне внешний интерфейс 962 может быть связан с процессором 952. Внешний интерфейс 962 может обеспечивать, например, проводную связь в некоторых вариантах осуществления или беспроводную связь в других вариантах осуществления, причем могут быть использованы несколько интерфейсов.

Память 964 сохраняет информацию в вычислительном устройстве 950. Память 964 может быть реализована в виде одного или нескольких считываемых компьютером носителей информации, энергозависимого блока или блоков памяти или энергонезависимого блока или блоков памяти. Также может быть предусмотрена память 984 расширения, которая может быть подключена к устройству 950 через интерфейс 982 расширения, который может включать в себя, например, интерфейс карты SIMM (Single In Line Memory Module - Однорядный модуль памяти). Такая память 984 расширения может обеспечивать дополнительное пространство для сохранения данных в устройстве 950, а также сохранять приложения или другую информацию для устройства 950. Прежде всего, память 984 расширения может содержать команды для выполнения или дополнения описанных выше процессов, а также может содержать защищенную информацию. Так, например, память 984 расширения может выступать в качестве модуля безопасности для устройства 950, и может быть запрограммирована с помощью команд, позволяющих безопасно использовать устройство 950. Кроме того, защищенные приложения могут быть предоставлены посредством SIMM-карты совместно с дополнительной информацией, например, путем размещения идентифицирующей информации на SIMM-карте в невзламываемом виде.

Память может включать в себя, например, флэш-память и/или NVRAM-память (Энергонезависимое запоминающее устройство с произвольной выборкой), о чем будет сказано ниже. В одном из вариантов осуществления

компьютерный программный продукт имеет материальное воплощение на носителе информации. Компьютерный программный продукт содержит команды, которые при выполнении осуществляют один или несколько способов, таких как описанные выше. Носитель информации представляет собой считываемый компьютером или машиночитаемый носитель, такой как память 964, память 984 расширения или память на процессоре 952, доступ к которому может быть получен, например, через приемопередатчик 968 или внешний интерфейс 962.

Устройство 950 может осуществлять беспроводную связь через интерфейс 966 связи, который при необходимости может включать в себя схемы цифровой обработки сигнала. Интерфейс 966 связи может обеспечивать связь в различных режимах или протоколах, таких как, среди прочего, голосовые вызовы GSM, обмен сообщениями SMS, EMS или MMS, CDMA, TDMA, PDC, WCDMA, CDMA2000 или GPRS. Такая связь может осуществляться, например, посредством радиочастотного приемопередатчика 968. Кроме того, возможна связь на коротких расстояниях, например, с помощью Bluetooth, WiFi или другого подобного приемопередатчика (не показан). Кроме того, приемный модуль 980 GPS (Global Positioning System) может предоставлять устройству 950 дополнительные данные, связанные с навигацией и местоположением, которые могут быть использованы работающими на устройстве 950 приложениями.

Устройство 950 может также осуществлять звуковую связь с помощью аудиокодека 960, который может принимать устную информацию от пользователя и преобразовывать ее в пригодную для использования цифровую информацию. Аудиокодек 960 может также генерировать звуковой сигнал для пользователя, например, посредством динамика, например, в трубке устройства 950. Такой звук может включать в себя звук голосовых телефонных вызовов, записанный звук (например, голосовые сообщения, музыкальные файлы и тому подобное), а также звук, генерируемый работающими на устройстве 950 приложениями.

Вычислительное устройство 950 может быть реализовано в различных видах, как показано на чертеже. Например, оно может быть реализовано в виде сотового телефона 980. Оно также может быть реализовано как часть смартфона 982, карманного персонального компьютера или другого подобного мобильного устройства.

Различные варианты осуществления описанных здесь систем и способов могут быть реализованы в цифровых электронных схемах, интегральных схемах, специально разработанных ASIC (Application Specific Integrated Circuits - Специализированных интегральных схемах), компьютерном оборудовании, микропрограммах, программном обеспечении и/или их комбинациях. Эти различные варианты осуществления могут включать в себя вариант осуществления в одной или нескольких компьютерных программах, исполняемых и/или интерпретируемых на программируемой системе, включающей в себя по меньшей мере один программируемый процессор, который может быть выполнен специализированным или общего назначения, и который соединен для приема данных и команд из системы хранения и для передачи данных и команд в систему хранения, по меньшей мере одно устройство ввода и по меньшей мере одно устройство вывода.

Эти компьютерные программы (также известные как программы, программное обеспечение, программные приложения или код) включают в себя машинные команды для программируемого процессора и могут быть реализованы на процедурном и/или объектно-ориентированном языке программирования высокого уровня, и/или на языке ассемблера/машинном языке. В настоящем документе термины «машиночитаемый носитель» и «компьютерный носитель» относятся к любому программному продукту, аппарату и/или устройству (например, магнитным дискам, оптическим дискам, памяти, программируемым логическим устройствам (ПЛИС)), используемым для предоставления машинных команд и/или данных программируемому процессору, включая сюда машиночитаемый носитель, который принимает машинные команды в виде машиночитаемого сигнала. Термин «машиночитаемый сигнал» относится к любому сигналу, используемому для предоставления машинных команд и/или данных программируемому процессору.

Для обеспечения взаимодействия с пользователем описанные здесь системы и способы могут быть реализованы на компьютере, имеющем устройство отображения (например, ЭЛТ (катодно-лучевую трубку) или ЖК (жидкокристаллический дисплей)) для вывода информации пользователю, а также клавиатуру и указывающее устройство (например, мышь или шаровой указатель), с помощью которых пользователь может вводить данные в

компьютер. Для взаимодействия с пользователем могут быть использованы и другие устройства, например, обратная связь с пользователем может иметь произвольную форму (например, визуальную, слуховую или тактильную), также и ввод от пользователя может быть осуществлен в произвольной форме, включая сюда акустическую, речевую или тактильную.

Описанные здесь системы и способы могут быть реализованы в вычислительном устройстве, включающем в себя внутренний компонент (например, сервер данных), промежуточный компонент (например, сервер приложений), фронтальный компонент (например, клиентский компьютер с графическим интерфейсом пользователя или Web-браузер, посредством которого пользователь может взаимодействовать с осуществлением описанных здесь систем и способов), или любую комбинацию таких внутренних, промежуточных или фронтальных компонентов. Компоненты системы могут быть соединены между собой с помощью любой формы или среды передачи цифровых данных (например, коммуникационной сети). Примерами коммуникационных сетей могут служить локальная сеть (LAN), глобальная сеть (WAN) и Интернет.

Вычислительное устройство может включать в себя клиенты и серверы. Клиент и сервер, как правило, удалены друг от друга и обычно взаимодействуют через сеть связи. Отношения клиента и сервера возникают в силу того, что компьютерные программы подлежат выполнению на соответствующих компьютерах и находятся друг с другом в отношениях клиент-сервер.

Описан ряд вариантов осуществления изобретения. Тем не менее, следует понимать, что в них могут быть внесены различные изменения, не выходящие за рамки духа и объема изобретения.

Кроме того, для достижения желаемых результатов изображенных на чертежах логических потоков не требуется конкретного порядка следования или последовательности. Кроме того, в описанных потоках некоторые шаги могут быть исключены, а другие - введены, а в описанных системах некоторые компоненты могут быть исключены, а другие - добавлены. Соответственно, другие варианты осуществления изобретения входят в объем приведенной ниже формулы изобретения.

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Компьютерно-реализуемый способ (400) обучения нейронной сети (373), причем сеть (373) обучают для последующей обработки многомерных временных рядов (223) измерений, которые содержат данные измерений, представляющие отдельные параметры (183) технологического процесса производственной установки (123) и, таким образом, для получения индикатора (Z') параметра для другого параметра (173, z) технологического процесса производственной установки (123), причем способ (400) включает в себя:

прием (411) первого содержащего исторические данные измерений многомерного временного ряда (211, 221) измерений от первой базовой установки (111) и второго содержащего исторические данные измерений многомерного временного ряда (221) измерений от физически другой второй базовой установки (121),

получение путем обработки первого и второго многомерных временных рядов измерений набора правил (251/252) преобразования так, что правила (251/252) преобразования позволяют преобразовательному модулю (311/321) преобразовывать соответственно первый и второй многомерные временные ряды (211, 221) измерений в первый и второй многомерные временные ряды (231, 241) признаков, причем многомерные временные ряды признаков являются инвариантными к относящимся к областям значений различиям первой и второй базовых установок (111, 121),

преобразование (452) первого многомерного временного ряда (211, 221) измерений в первый многомерный временной ряд (232) признаков посредством применяющего правила (251/252) преобразования преобразовательного модуля (322),

прием (462) содержащего данные (Z) измерений одномерного временного ряда (272) измерений другого параметра (173, z) технологического процесса от первой базовой установки (112), и

обучение (472) нейронной сети (372) с использованием первого многомерного временного ряда (232) признаков на входе и с использованием одномерного временного ряда (272) измерений на выходе.

2. Способ (400) по п. 1, причем получение (421) набора правил (251/252) преобразования осуществляет компьютер, который итеративно выполняет следующие действия:

• преобразование (431) первого многомерного временного ряда (211) измерений в первый многомерный временной ряд (231) признаков и преобразование (431) второго многомерного временного ряда (221) измерений во второй многомерный временной ряд (241) признаков с использованием набора правил предварительного преобразования,

• различение (441) посредством дискриминирующего модуля (361) первого и второго временных рядов (231, 241) признаков как происходящих из первого и второго многомерных временных рядов (211, 221) измерений, и

• изменение (451) набора правил предварительного преобразования до тех пор, когда дискриминирующий модуль (361) более не может различать происхождение (1, 2) данных.

3. Способ по п. 2, причем дискриминирующий модуль (361) определяет, что более не может различать происхождение данных путем работы до тех пор, когда состязательные потери достигнут максимального значения.

4. Способ по п. 2, причем шаги получения (421) набора правил преобразования и обучения (472) нейронной сети (372) выполняют итеративно, причем преобразовательный модуль (311/321) дополнительно принимает создаваемые нейронной сетью (372) в процессе обучения сетевые потери (LOSS) так, что итеративное выполнение останавливают при достижении сетевыми потерями минимального уровня.

5. Способ (400) по одному из предшествующих пунктов, причем комбинация преобразовательного модуля (311/321) и дискриминирующего модуля работает (361) как порождающая состязательная сеть.

6. Способ (400) по одному из предшествующих пунктов, причем шаги способа, такие как прием (411) многомерных временных рядов (211, 221) измерений, получение (421) набора правил преобразования, преобразование (452) и обучение (472) нейронной сети, выполняют для многомерных временных

рядов измерений, для которых посредством модуля (383) селекции данных переменные были определены как связанные с другим параметром (173, z) технологического процесса.

7. Способ (400) по п. 6, причем модуль (383) селекции данных обучают посредством модуля (380) обучения селектора, который побуждает компьютер к итеративному выполнению шагов способа по получению, преобразованию и обучению, при этом критерием является минимизация функции потерь в процессе обучения (482) нейронной сети.

8. Способ (400) по одному из предшествующих пунктов, причем прием (411) содержащих исторические данные первого и второго многомерных временных рядов (211, 221) измерений выполняют после приспосабливания вида данных для данных измерений.

9. Способ (400) по п. 8, причем приспосабливание вида данных выполняют с помощью шагов, выбранных из следующих: анализ содержащих образцы звука данных измерений путем классификации образцов звука по категориям, и анализ содержащих образы данных измерений путем классификации образов по категориям.

10. Способ (400) по одному из п.п. 8 или 9, причем приспосабливание выполняют итеративно для различных целей, причем цели выбирают по критерию минимизации функции потерь в процессе обучения (482) нейронной сети.

11. Способ (400) по одному из предшествующих пунктов, причем нейронная сеть (372) представляет собой регрессионную нейронную сеть.

12. Способ (400) по одному из предшествующих пунктов, причем данные измерений связаны с параметрами технологического процесса, которые представляют собой физические свойства, выбранные из следующих: температура, давление, химический состав материалов внутри установки,

внешний вид части производственной установки, и звук, издаваемый производственной установкой.

13. Способ (400) по одному из предшествующих пунктов, причем производственные установки представлены химическими реакторами, металлургическими печами, технологическими емкостями, и двигателями.

14. Компьютерно-реализуемый способ (500) использования нейронной сети (373), которая предварительно обучена согласно способу (400) по одному из п.п. 1-13, причем способ (500) использования включает в себя следующие шаги:

прием (513) содержащего данные измерений для наблюдаемой производственной установки (123) многомерного временного ряда (223) измерений,

преобразование (523) многомерного временного ряда (223) измерений в многомерный временной ряд (243) признаков, и

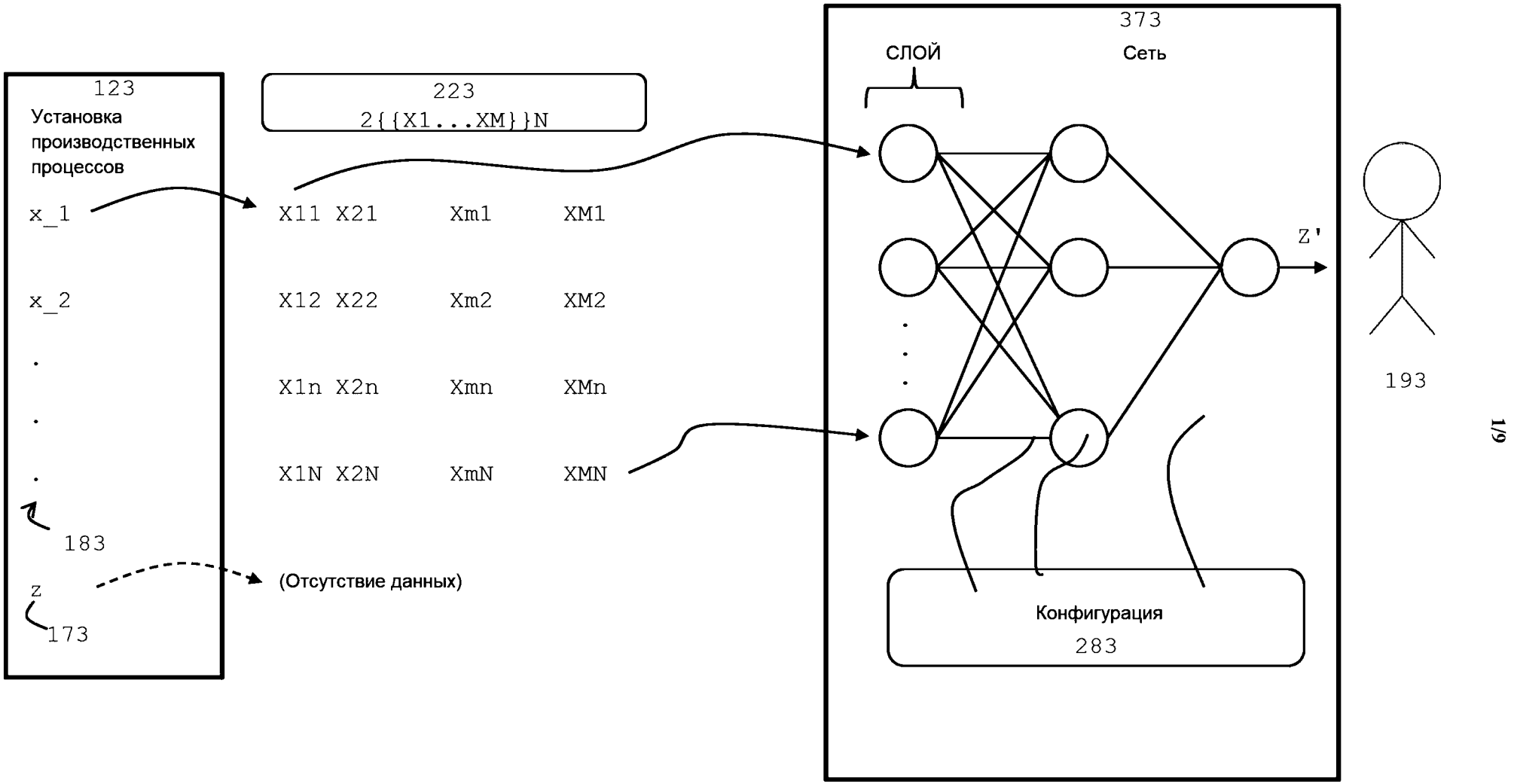
использование (533) нейронной сети (373) для получения индикатора (Z') параметра.

15. Способ (500) по п. 14, причем использование нейронной сети происходит в ситуациях, выбранных из следующего: отказ датчика, обрыв линии передачи данных, минимизация участия человека при измерении.

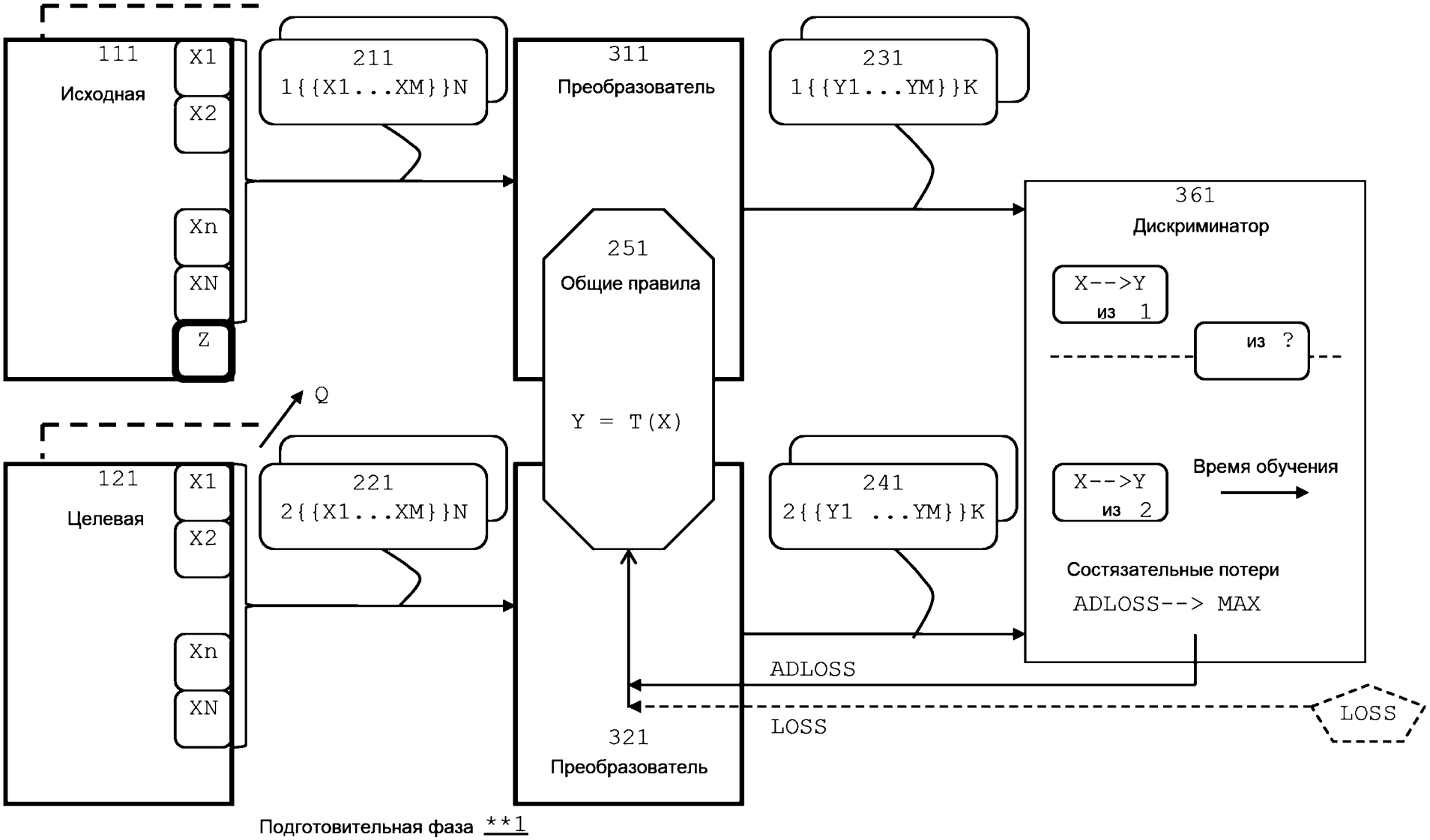
16. Компьютерный программный продукт, который, будучи загруженным в память компьютерной системы и исполняемым посредством по меньшей мере одного процессора компьютерной системы, побуждает компьютерную систему к выполнению шагов компьютерно-реализуемого способа согласно любому из предшествующих пунктов.

17. Компьютерная система, которая включает в себя несколько компьютерно-реализуемых модулей, которые при выполнении компьютерной системой могут выполнять шаги компьютерно-реализуемого способа по одному из п.п. 1-15.

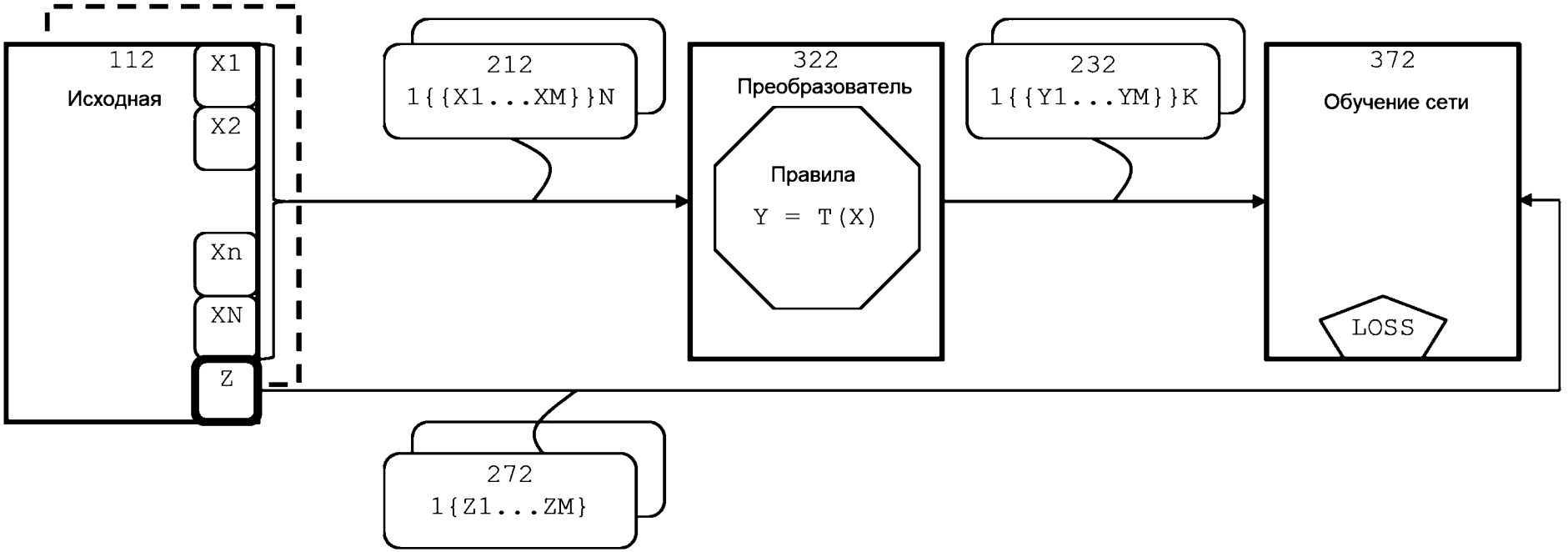
18. Применение выполняющего способ по одному из п.п. 14-15 компьютера для получения индикатора (Z') параметра, который представляет параметр (z) производственной установки (123).


123
Установка производственных процессов
x_1
x_2
183
z
173
223
2{{X1...XM}}N
X11 X21 Xm1 XM1
X12 X22 Xm2 XM2
X1n X2n Xmn XMn
X1N X2N XmN XMN
(Отсутствие данных)
373
СЛОЙ
Сеть
Z'
193
Конфигурация
283

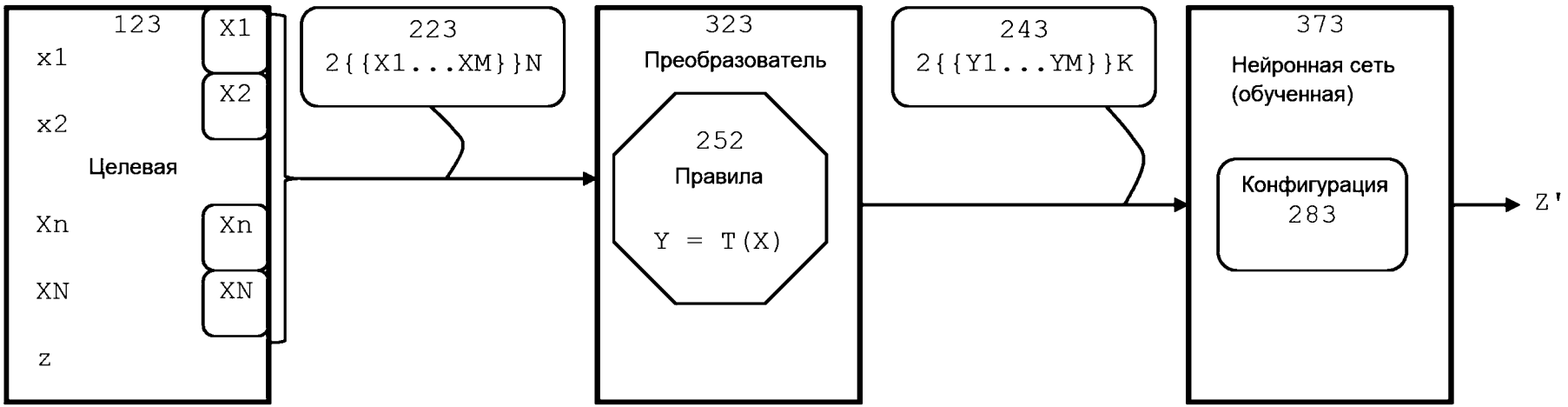
Фиг. 1


111
Исходная
X1
X2
Xn
XN
Z
211
1{{X1...XM}}N
311
Преобразователь
231
1{{Y1...YM}}K
361
Дискриминатор
251
Общие правила
Y = T(X)
X-->Y
из 1
из ?
Q
121
Целевая
X1
X2
Xn
XN
221
2{{X1...XM}}N
241
2{{Y1 ...YM}}K
Время обучения
X-->Y
из 2
Состязательные потери
ADLOSS--> MAX
ADLOSS
LOSS
LOSS
321
Преобразователь
Подготовительная фаза **1

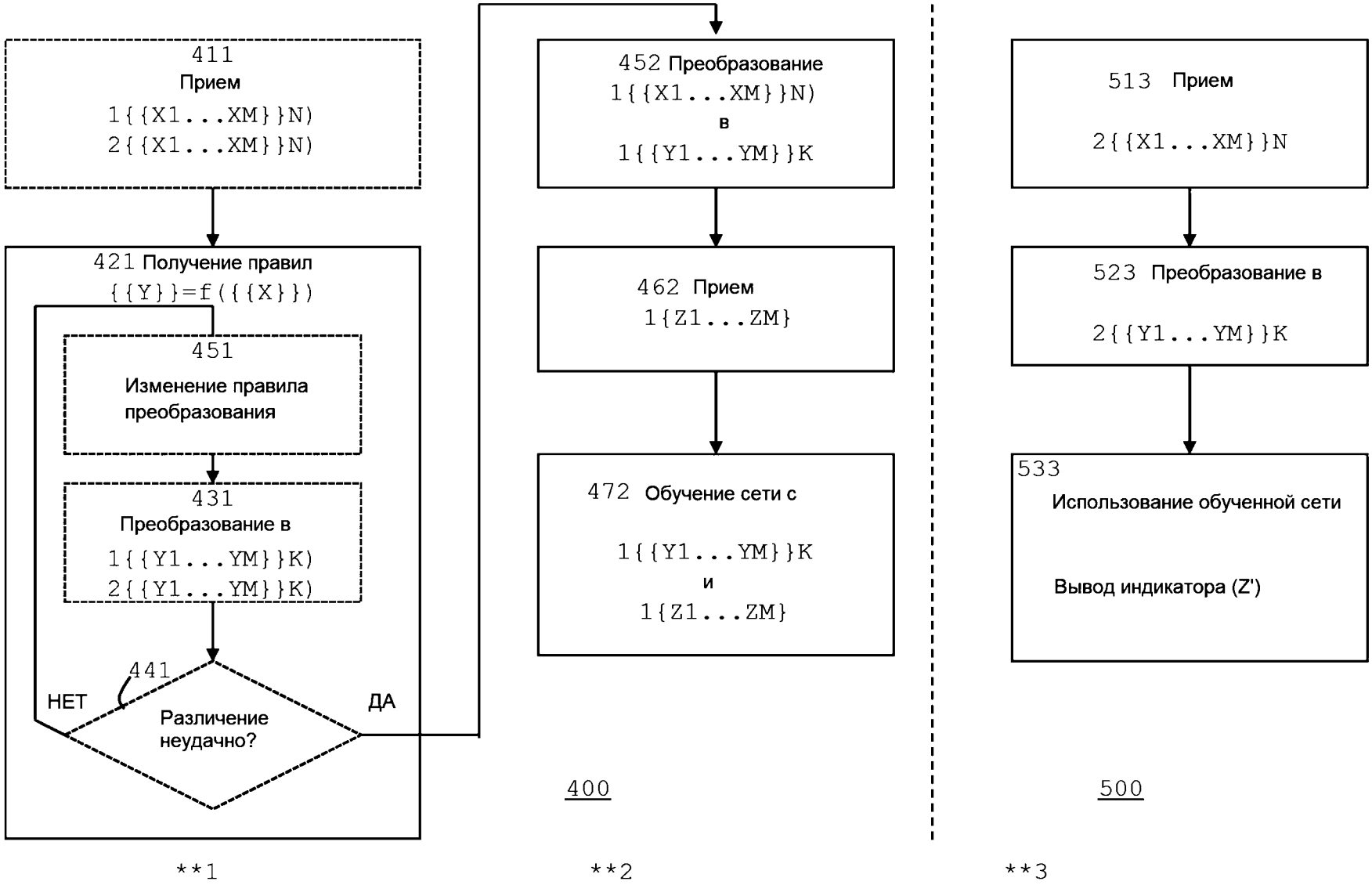
Фиг. 2


112
Исходная
X1
X2
Xn
XN
Z
212
1{{X1...XM}}N
322
Преобразователь
Правила
Y = T(X)
232
1{{Y1...YM}}K
372
Обучение сети
LOSS
272
1{Z1...ZM}

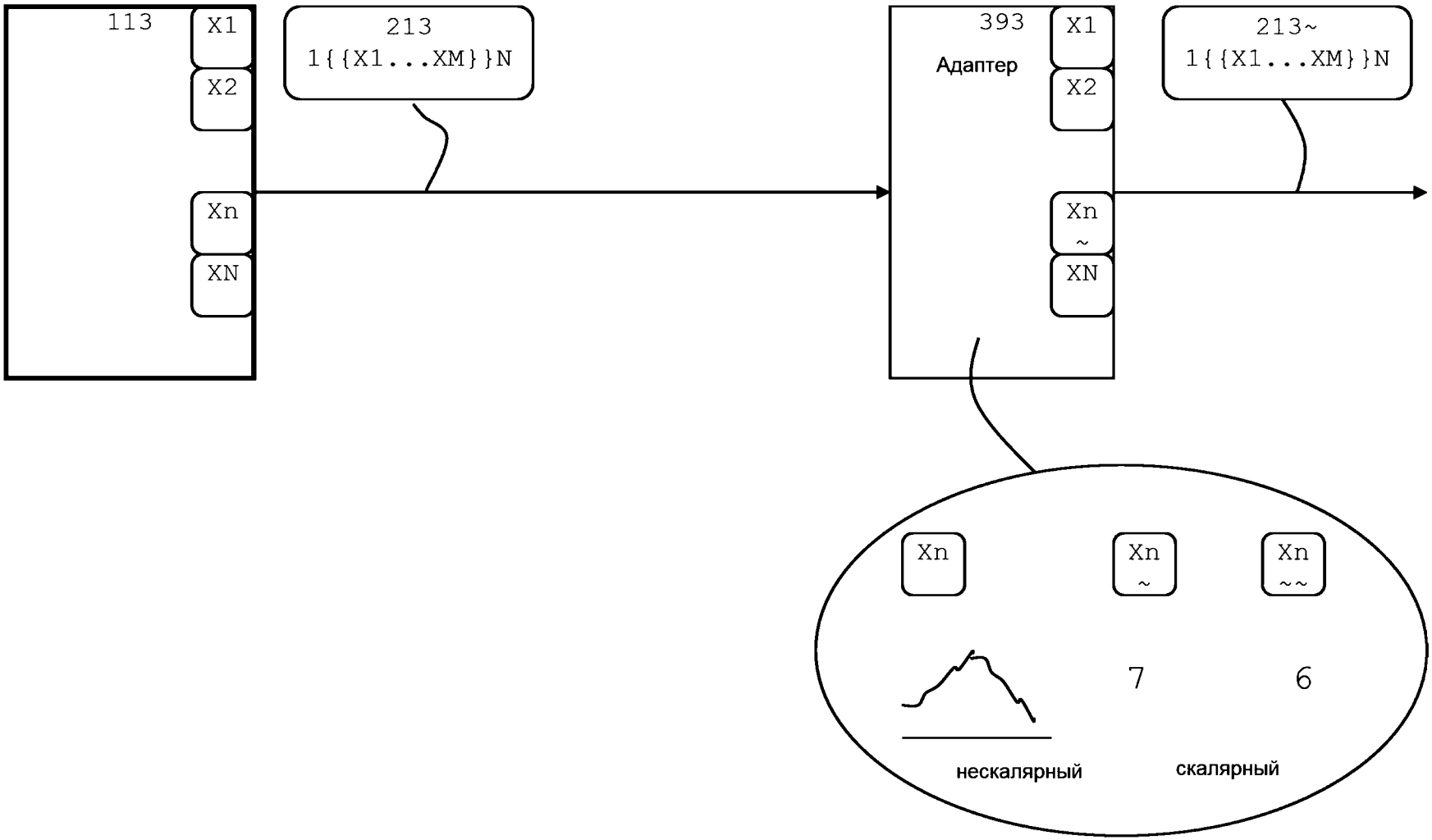
Фиг. 3


123
x1
x2
Целевая
Xn
XN
z
X1
X2
Xn
XN
223
2{{X1...XM}}N
323
Преобразователь
252
Правила
Y = T(X)
243
2{{Y1...YM}}K
373
Нейронная сеть
(обученная)
Конфигурация
283
Z'
**3

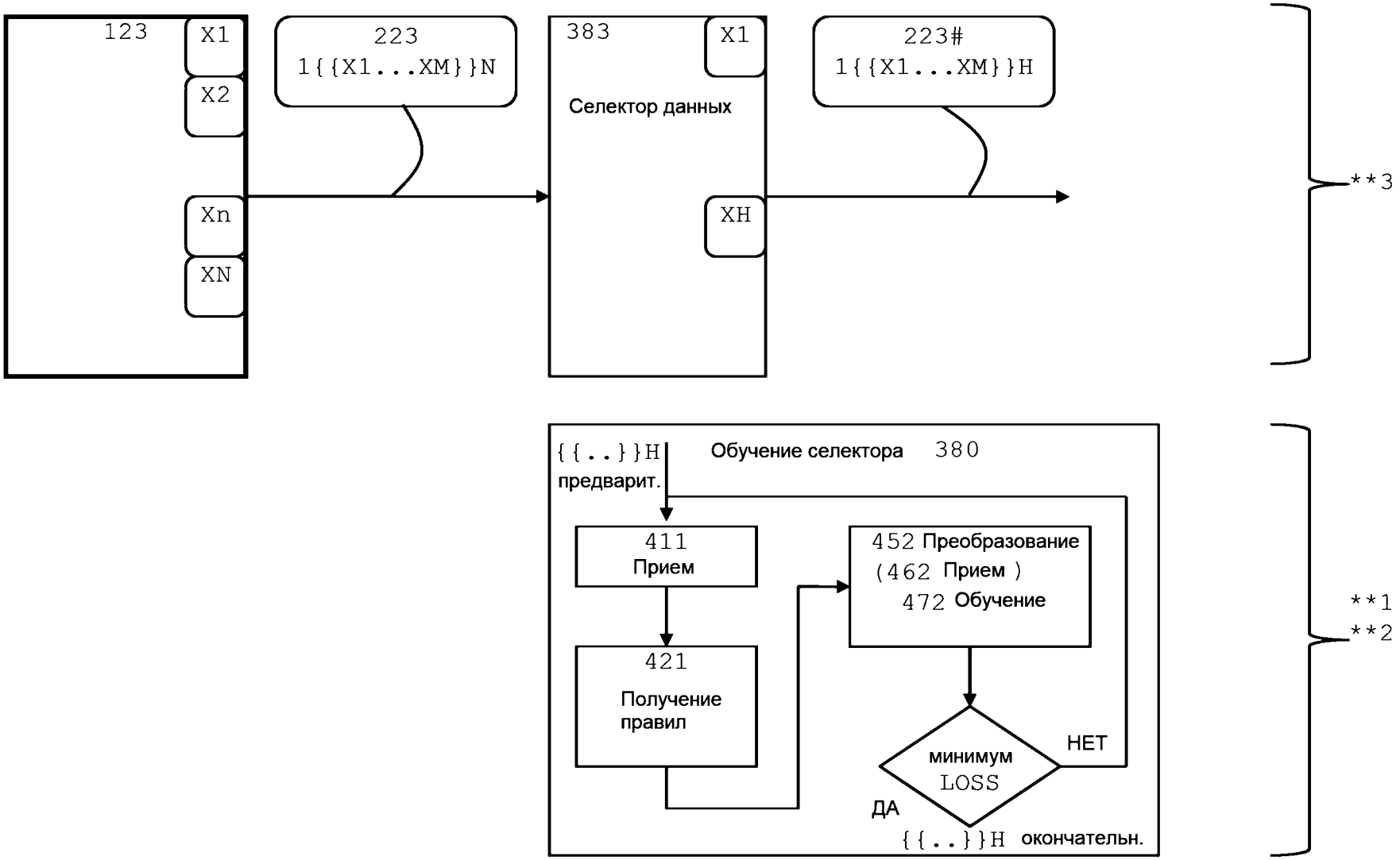
Фиг. 4


411
Прием
1{{X1...XM}}N)
2{{X1...XM}}N)
421 Получение правил
{{Y}}=f({{X}})
451
Изменение правила преобразования
431
Преобразование в
1{{Y1...YM}}K)
2{{Y1...YM}}K)
441
НЕТ
ДА
Различение неудачно?
452 Преобразование
1{{X1...XM}}N)
в
1{{Y1...YM}}K
462 Прием
1{Z1...ZM}
472 Обучение сети с
1{{Y1...YM}}K
и
1{Z1...ZM}
513 Прием
2{{X1...XM}}N
523 Преобразование в
2{{Y1...YM}}K
533
Использование обученной сети
Вывод индикатора (Z')
400
500
**1
**2
**3

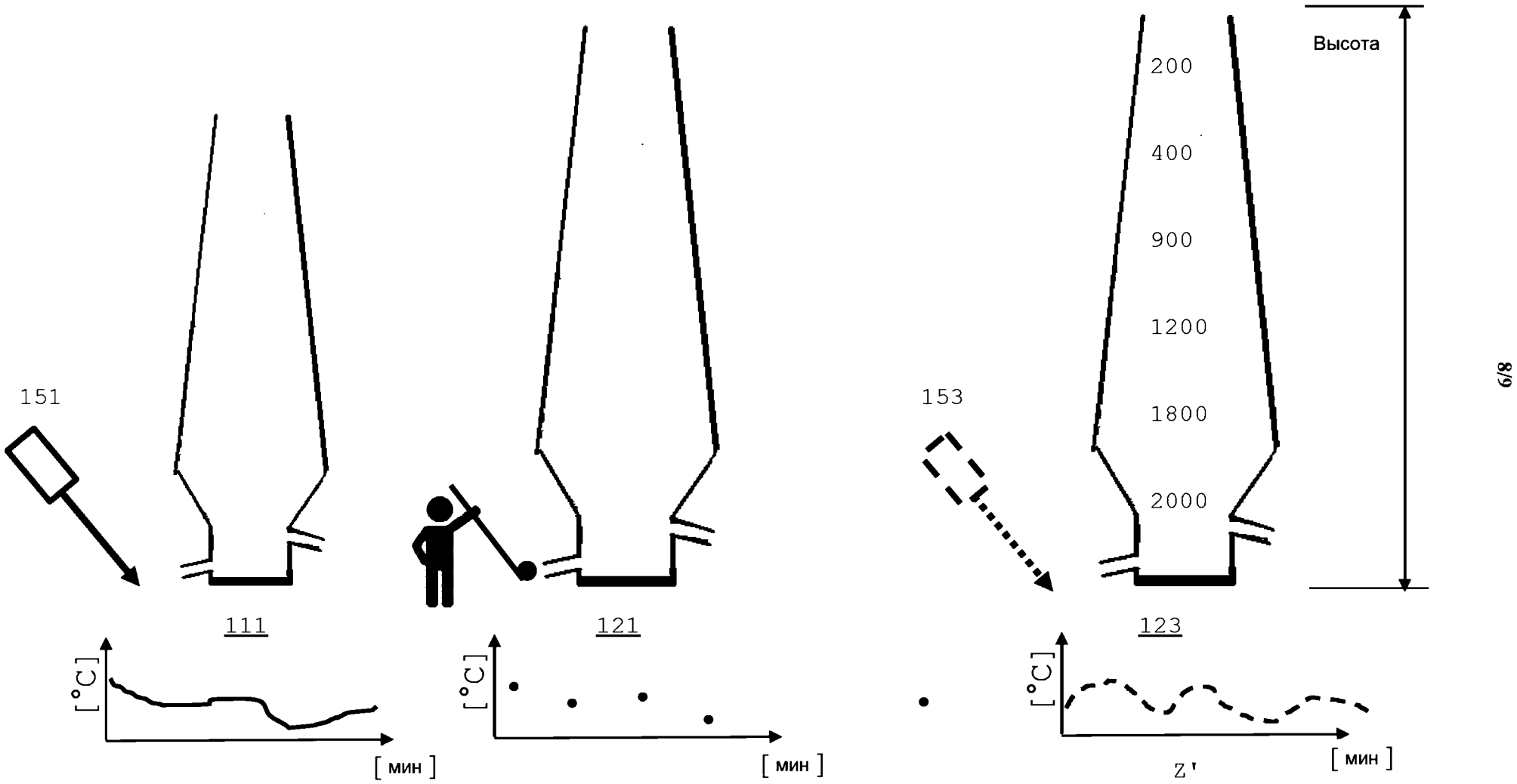
Фиг. 5


113
X1
X2
Xn
XN
213
1{{X1...XM}}N
393
Адаптер
X1
X2
Xn
~
XN
213~
1{{X1...XM}}N
Xn
Xn
~
Xn
~~
7
6
нескалярный
скалярный

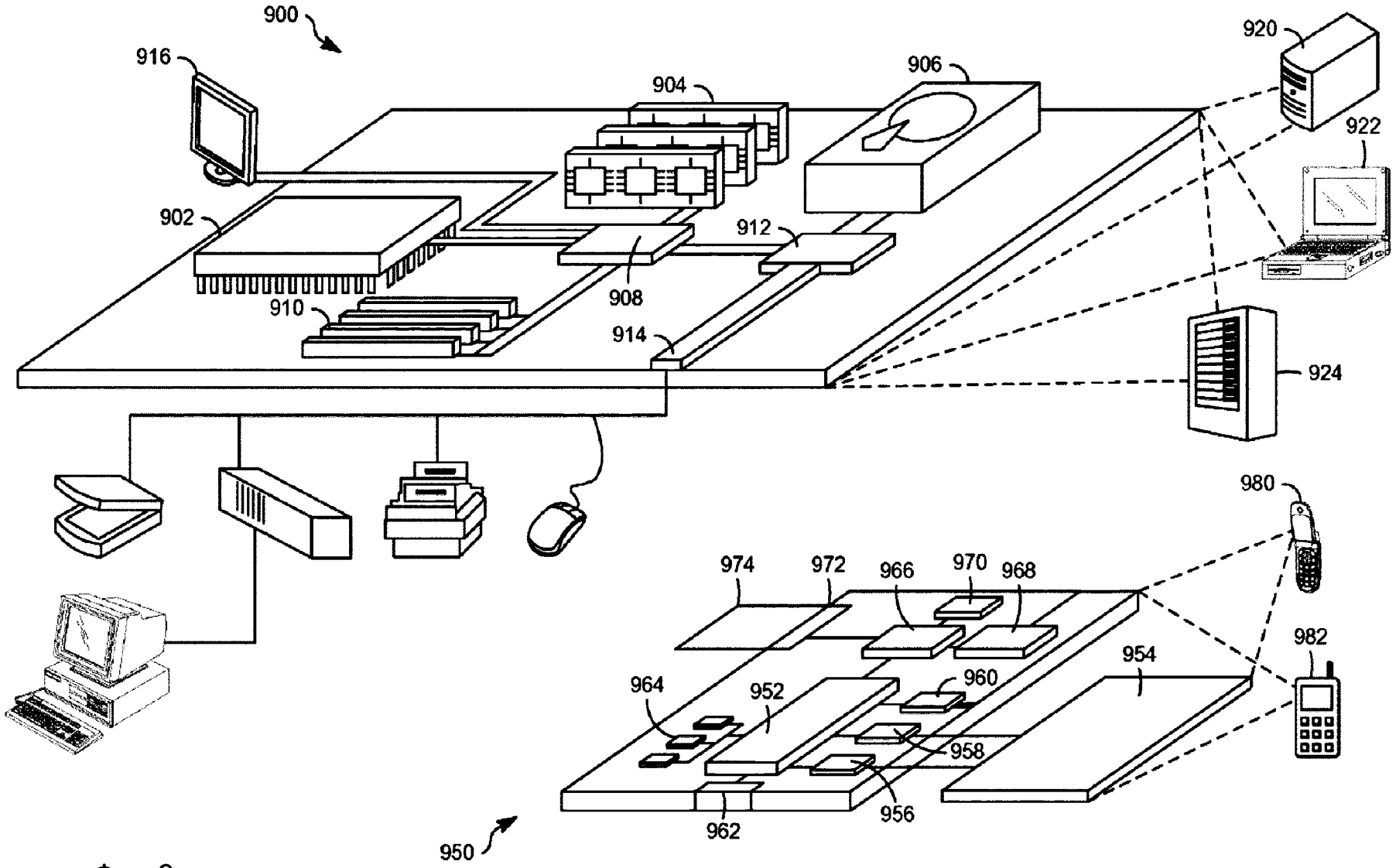
Фиг. 6


123
X1
X2
Xn
XN
223
1{{X1...XM}}N
383
X1
Селектор данных
XH
223#
1{{X1...XM}}H
**3
{{..}}H
предварит.
Обучение селектора 380
411
Прием
421
Получение
правил
452 Преобразование
(462 Прием)
472 Обучение
минимум
LOSS
НЕТ
ДА
{{..}}H окончательн.
**1
**2

Фиг. 7


151
111
[°C]
[мин]
121
[°C]
[мин]
153
200
400
900
1200
1800
2000
Высота
123
[°C]
[мин]
Z'

Фиг. 8


900
916
904
906
920
922
902
912
908
910
914
924
980
974
972
966
970
968
982
954
964
952
960
958
956
962
950

Фиг. 9