

Платформа-ГНС

Унифицированная программная платформа машинного обучения



«Платформа-ГНС» – это отечественное программное обеспечение, реализующее полный стек технологий, необходимых для создания конечных прикладных решений на основе глубоких нейронных сетей (ГНС).

Это интегрированная программная среда, которая обеспечивает законченный цикл разработки систем искусственного интеллекта на основе ГНС на всех этапах от создания данных для обучения до автоматизированного тестирования и портирования обученных нейросетевых моделей на выбранные аппаратные платформы.

«Платформа-ГНС» содержит готовые типовые решения для задач компьютерного зрения и анализа сигналов, таких как обнаружение объектов на изображениях и видеопоследовательностях и многоспектральных данных, дешифрирование, классификация, семантическая сегментация, сопровождение объектов, обработка и комплексирование изображений различных диапазонов, устранение шумов и помех, устранение смаза и расфокусировки и др.

«Платформа-ГНС» отличается поддержкой отечественных ОС, использованием единого унифицированного формата хранения моделей, совместимостью с основными средами разработки нейросетевых моделей, возможностью портирования созданных алгоритмов на отечественные аппаратные платформы, возможностью решения типовых задач средствами визуального программирования.

На уровне работы с типовыми решениями «Платформа-ГНС» дает возможность специалистам в прикладных областях без дополнительных специальных знаний в области машинного обучения решить свои прикладные задачи и использовать полученные конечные решения в составе существующих или перспективных систем.

В то же время, на уровне работы с программным кодом «Платформа-ГНС» предоставляет специалистам в области машинного обучения удобный инструментарий для встраивания и разработки новых решений, которые далее могут использоваться специалистами-прикладниками как готовые типовые.

Описанные далее основные характеристики, элементы и функции «Платформы-ГНС» относятся к текущей версии 1.8. Также представлены новые возможности «Платформы-ГНС», разработанные для реализации в следующей версии 2.0 (версия «Автомат»), релиз которой запланирован на июль 2023 г. Новые элементы и функции Платформы 2.0 были впервые представлены и продемонстрированы на семинаре-конференции «PLAT-2023», прошедшем в ФАУ «ГосНИИАС» 15 февраля 2023 г.

Содержание

Состав и основные характеристики «Платформы-ГНС»

«Платформа-ГНС» — это ... Преимущества работы в «Платформе-ГНС» Сервер «Платформы-ГНС» Клиент «Платформы-ГНС»

Новые функции в составе «Платформы-ГНС»

Автоматический подбор архитектур и параметров обучения Редактор выборок и аугментаций Средства поддержки пользователей

Справочная информация

Системные требования «Платформы-ГНС» Типы задач и поддерживаемые архитектуры нейронных сетей Фреймворки машинного обучения и библиотека PlatLib

Состав и основные характеристики «Платформы-ГНС»

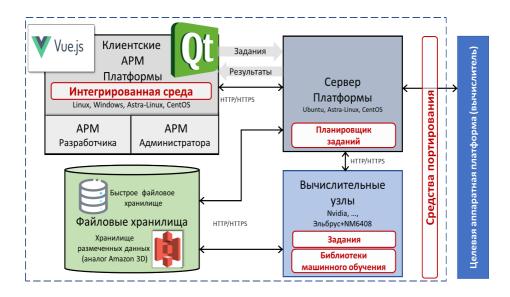


«Платформа-ГНС» — это интегрированная программная среда. Она обеспечивает полный цикл разработки решений на основе ГНС, включая работу с данными и разметку данных, формирование архитектуры и обучение ГНС, тестирование и портирование обученных моделей на целевые аппаратные платформы.

«Платформа-ГНС» представляет собой клиент-серверное программное обеспечение, в котором клиентская часть устанавливается на удалённых рабочих компьютерах, а серверная – на специализированных высокопроизводительных компьютерах, кластерах в локальной сети, либо в виде облачного решения на специализированных площадках организаций.

«Платформа-ГНС» использует технологию «тонкого клиента», при которой практически все ресурсоемкие вычислительные операции проводятся удалённо серверной частью. Такая архитектура обусловлена, прежде всего, спецификой задачи, которая подразумевает проведение крайне ресурсоемких параллельных вычислений с использованием больших (сотни терабайт) объёмов данных.

При такой организации вся информация, включая данные проекта, исходные коды, обучающие и тестовые выборки, хранится в защищённом виде на удалённом сервере. В клиентской части хранятся временные данные, необходимые для ускорения работы системы. После завершения



работы временные файлы удаляются из системы. Это позволяет реализовать концепцию командной работы над проектом и решение сложных задач коллективом разработчиков, а также осуществлять мониторинг проектной деятельности по каждому проекту.

Преимущества работы в «Платформе-ГНС»

- Динамически расширяемый набор готовых алгоритмов обучения и дообучения ГНС для различных прикладных задач;
- Возможность автоматического подбора параметров обучения и архитектур ГНС;
- Наличие прикладных пакетов по работе с изображениями, одномерными и многомерными сигналами, таблицами и т.д.;
- Поддержка отечественного фреймворка машинного обучения PlatLib;
- Поддержка портирования обученных решений на отечественные процессоры;
- Возможность работы со всеми известными фреймворками машинного обучения;
- Сертифицируемый на отсутствие недокументированных возможностей исходный код.

Сервер «Платформы-ГНС»

Серверная часть «Платформы-ГНС» выполняет роль **оркестратора** всей системы и включает следующие элементы:

- 1. Программный модуль реализации логики работы, хранения данных и API;
- 2. Программный модуль взаимодействия с вычислительным кластером;
- 3. Программный модуль управления объектным хранилищем;
- 4. База данных аннотированных эталонных данных для обучения нейросетевых алгоритмов;
- 5. База данных проектов и моделей.

Клиент «Платформы-ГНС»

Клиентская часть «Платформы-ГНС» устанавливается на пользовательские рабочие места. «Платформа-ГНС» поддерживает работу трех категорий пользователей:

- Разработчик специалист, решающий задачи подготовки данных, формирования и обучения нейронных сетей, портирования и выгрузки обученных моделей;
- Администратор проектов оператор, решающий задачи организации

- и управления проектной деятельностью внутри «Платформы-ГНС», включая создание проектов, организацию проектных команд, распределение ролей внутри команды (работа с разметкой данных, проектом);
- Администратор ресурсов оператор, решающий задачи системного администрирования программного обеспечения «Платформы-ГНС», включая управление серверной частью «Платформы-ГНС», организации вычислительных ресурсов кластера, а также организации загрузки данных в объектное хранилище «Платформы-ГНС».

Трем категориям пользователей соответствуют три различных клиентских приложения (автоматизированные рабочие места, APM): APM Разработчика, APM Администратора проектов, APM Администратора ресурсов. Клиент-серверная архитектура «Платформы-ГНС» позволяет устанавливать эти APM как на один компьютер, так и на различные компьютеры в одной локальной сети.

APM Разработчика поддерживает три уровня взаимодействия с пользователем для решения задач машинного обучения, реализованные в виде отдельных модулей:

- Модуль готовых решений;
- Модуль работы с исходным кодом;
- Модуль визуального программирования.

Модуль **готовых решений** является самым простым уровнем работы с «Платформой-ГНС». Благодаря интегрированному в «Платформу-ГНС» набору типовых решений (готовых архитектур нейронных сетей с параметрами настройки и обучения), данный уровень работы доступен практически любому пользователю без специального образования. Пользователю необходимо пройти 4 последовательных шага:

- **Шаг 1.** Выбрать из списка готовых решений подходящее типовое решение для своей прикладной задачи.
- **Шаг 2.** С помощью встроенного в «Платформу-ГНС» инструмента разметки данных подготовить и разметить выборки для обучения и тестирования.
- **Шаг 3.** В выбранном типовом решении настроить основные параметры, запустить обучение, а затем (если требуется) тестирование обученной модели.
- **Шаг 4.** Выгрузить обученную нейросетевую модель в формате ONNX, либо портировать ее на одну из поддерживаемых «Платформой-ГНС» аппаратных платформ для дальнейшего использования в конечном прикладном решении.



Модуль работы с исходным кодом рассчитан на специалистов в области обучения нейронных сетей, которые способны написать код нового нейросетевого решения, либо встроить (интегрировать) в «Платформу-ГНС» ранее разработанный сторонний код, после чего отладить его в специализированном отладчике. В «Платформе-ГНС» для этого используется интегрированный внешний отладчик Jupyter – интерактивная среда разработки и отладки кода. На этапе написания, встраивания и отладки кода пользовательские решения могут непосредственно запускаться на вычислительных узлах сервера «Платформы-ГНС». После завершения разработки, пользовательские решения можно перенести в список готовых решений. При создании проекта на основе данного решения пользователю будет доступен весь функционал, как и для работы с типовыми решениями. Для этого предусмотрены процедура экспорта написанного кода, а также специальная библиотека программного интерфейса «Платформы-ГНС» — PlatformAPI, которая позволяет использовать унифицированные форматы и протоколы передачи данных между клиентской и серверной частями «Платформы-ГНС», а также между хранилищами данных и вычислительными узлами.

Модуль визуального программирования предназначен для интерактивного формирования схем обработки и анализа данных. Его применение в большей степени ориентировано на задачи обучения будущих специалистов в области анализа данных. Работа с ним требует обладания знаниями о построении типовых схем («пайплайнов») построения модульных процедур анализа данных. Функционал модуля даёт возможность создания пользовательских схем, которые могут объединяться в более крупные компоненты — суперблоки и связываться между собой логическими переходами. Процесс обучения нейросети, как и в модуле готовых решений, может отслеживаться с помощью различных средств визуализации в реальном времени.

Новые функции «Платформы-ГНС» версии 2.0 («Автомат»)

Автоматический подбор архитектур и параметров обучения

Начинающему пользователю сложно сориентироваться в много образии архитектур и выбрать правильную архитектуру нейронной сети для решения конкретной прикладной задачи. Также непросто для заданной архитектуры, конкретной задачи и обучающей выборки оптимально (с точки зрения заданного критерия) настроить параметры обучения.

В качестве средства поддержки пользователей, кардинально упрощающего работу по обучению нейросетей, в «Платформе-ГНС», начиная с версии 2.0 («Автомат»), будут встроены механизмы автоматического подбора архитектур и параметров обучения под задачи, выборки и критерии.

Для подбора архитектуры пользователь должен выбрать класс решаемой задачи и загрузить в «Платформу-ГНС» размеченные данные. Программное обеспечение осуществит автоматический подбор архитектуры, исходя из обеспечения максимального качества конечного результата по выбранному критерию. Автоматически сформированная архитектура далее может быть включена в список архитектур, доступных в готовом решении, и после этого использоваться в других задачах проекта для обучения на новых данных.

Подбор параметров обучения часто необходим для тех случаев, когда пользователь имеет точное представление об архитектуре сети, на которой он хочет получить конечное решение, но испытывает трудности с определением значений тех или иных параметров обучения. Встроенный механизм автоматического подбора параметров поможет подобрать оптимальное значение параметров обучения, основываясь на размеченной обучающей выборке и заданном критерии качества.

Редактор выборок и аугментаций

При обучении нейросетевых алгоритмов анализа данных часто требуется обеспечить устойчивость результатов работы обученной нейросети к тем или иным типам преобразований. Например, распознавание объектов должно быть устойчиво к сдвигам, поворотам, масштабированию и подобным преобразованиям изображений. Для достижения такой устойчивости необходимо пополнить обучающие выборки большим количеством изображений исходной базы примеров, подвергнутых таким преобразованиям. В версии 2.0 («Автомат») в «Платформу-ГНС» для расширения состава и качества обучающих

выборок будет добавлен специальный программный модуль — **редактор** выборок и аугментаций.

Данный модуль позволяет формировать последовательности необходимых преобразований данных для автоматизированного применения этих преобразований к выборкам при обучении нейронных сетей.

Средства поддержки пользователей

Для облегчения процесса освоения работы с «Платформой-ГНС» и более быстрой адаптации пользователя в версии 2.0 («Автомат»), помимо существовавшего ранее механизма выдачи справок, появятся три типа подсказок:

- по работе с «Платформой-ГНС» (режим «рекомендации»);
- по последовательности запуска заданий (режим «инструкции»);
- по выбору стартовых параметров обучения и архитектуры (режим «база знаний»).

Справочная информация

Системные требования «Платформы-ГНС» Для запуска и использования «Платформы-ГНС» предъявляются следующие технические требования.



Клиент

| Минимальные требования | | |
|------------------------|---------------------------------------|--|
| Операционная | • Windows 7/10/11 (64 бит) | |
| система | • Linux версии не ниже 4.4 (64 бит) | |
| | • AstraLinux версии 1.7 (64 бит) | |
| | • CentOS (версия 8) | |
| Процессор | Процессор с тактовой частотой 1,2 ГГц | |
| ОЗУ | Не менее 4 Гбайт оперативной памяти | |
| Жесткий диск | 8 Гбайт свободного пространства на | |
| | жёстком диске (HDD) | |
| Разрешение | 1920x1080 (FullHD) | |
| монитора | | |
| Канал доступа к | 100 Мбит/с | |
| внешним сетям | | |

| Рекомендованные требования | | |
|----------------------------------|--|--|
| Процессор | Процессор с тактовой частотой 2 ГГц (или выше) | |
| ОЗУ | 16 Гбайт (или больше) оперативной памяти | |
| Жесткий диск | 16 Гбайт (или больше) свободного пространства на жёстком диске (HDD) | |
| Канал доступа к внешним сетям | 1 Гбит/с | |



Сервер

| Минимальные требования для каждого узла кластера | | |
|--|--|--|
| Операционная система | Ubuntu 18.04 или любой сопоставимый GNU/Linux дистрибутив | |
| Процессор | Процессор с 2 ядрами и тактовой частотой 2 ГГц | |
| ОЗУ | Не менее 32 Гбайт оперативной памяти | |
| GPU | 1 GPU ускоритель NVIDIA архитектуры Pascal (или выше) с объёмом памяти 8 Гбайт | |
| HDD | Высокопроизводительное сетевое файловое хранилище размером 1 Тбайт, подключенное к каждому узлу кластера | |
| Internet | Высокоскоростной канал связи между узлами кластера с пропускной способностью 1 Гбит/с | |

| Рекомендованные требования для каждого узла кластера | | |
|--|--|--|
| Процессор | Процессор с 8 (или более) ядрами с тактовой частотой 3 ГГц (или выше) | |
| ОЗУ | 128 Гбайт (или больше) оперативной памяти; | |
| GPU | 2 (или более) GPU ускорителя NVIDIA архитектуры Pascal (или выше) с объёмом памяти 12 Гбайт (или больше) | |
| HDD | Высокопроизводительное сетевое файловое хранилище размером 40 Тбайт (или больше), подключенное к каждому узлу кластера | |
| Internet | Высокоскоростной канал связи между узлами кластера на базе технологий Infini-Band, FiberChannel, 40GbE, 100GbE (либо аналог) | |

Возможна реализация с одноузловым сервером (без кластера). Требования аналогичны требованиям к узлам кластера.

Готовые решения и доступные архитектуры нейронных сетей

Готовые типовые решения «Платформы-ГНС» содержат более 50 различных архитектур нейронных сетей, предназначенных для основных типов задач машинного обучения. Для существенной части этих архитектур обеспечивается автоматизированное портирование на отечественные вычислители. Список поддерживаемых архитектур постоянно расширяется.

В таблице, представлены основные готовые решения и соответствующие доступные в них архитектуры нейронных сетей.

| Типовые решения | Архитектуры |
|----------------------|--|
| Классификация | Семейство ResNet: |
| изображений | *ResNet18, 34, 50, 101, 152 |
| | SE-ResNet, SE-ResNeXt, WideRes- |
| | Net50,101, ResNeXt |
| | Альтернатива ResNet: |
| | HRNetV2, EfficientNet, InceptionV4 |
| | Легковесные сети: |
| | *MobileNetV1,V2 |
| | Самые первые: |
| | *VGG11,13,26,16 |
| | SqueezeNet, *LeNet;* AlexNet |
| Семантическая | Атомарные: *Unet, HDSE_Unet |
| сегментация | Энкодеры: SeResNext50,101 |
| | MobilenetV2, ResNext101, WideResNet50, |
| | HRNetV2, Xception |
| | Декодеры: DeepLabV3, DeepLabV3+, |
| | SegAdpHead, SqRDU, LightHamHead |
| Обнаружение | Одноэтапный SSD детектор с ГНС для |
| объектов/обнаружение | извлечения признаков: *ResNet18,34,50, |
| по многоспектральным | *MobileNetV1,V2, *ShuffleNetV2, *Effi- |
| данным | cientNet |
| | Детектор семейства YOLO v4 |
| Ориентированное | BBAVectors |
| обнаружение | |

| Дешифрирование | Двухэтапный детектор Faster R-CNN |
|------------------------------------|---|
| Слежение за одним | SiamMask |
| объектом | |
| Слежение за | ByteTrack |
| несколькими | |
| объектами | |
| Обработка | *DnCNN, *Unet, *ResNet, *Pixel, |
| изображений: | *PatchGAN |
| • устранение смаза | |
| и расфокусировки | |
| (деблюринг) | |
| • устранение шумов и | |
| помех (денойзинг) | |
| • комплексирование | |
| Синтез изображений | Условный синтез |
| 1 | D' AD' C L CAN |
| | Схемы обучения: Pix2Pix, CycleGAN |
| | Схемы обучения: Pix2Pix, CycleGAN Архитектуры: Unet, ResNet, Pixel, Patch- GAN |
| | Архитектуры: Unet, ResNet, Pixel, Patch- GAN |
| | Архитектуры: Unet, ResNet, Pixel, Patch- |
| Обработка одномерных | Архитектуры: Unet, ResNet, Pixel, Patch- GAN Безусловный синтез |
| | Архитектуры: Unet, ResNet, Pixel, Patch-GAN Безусловный синтез ProGAN, StyleGAN |
| Обработка одномерных | Архитектуры: Unet, ResNet, Pixel, Patch-GAN Безусловный синтез ProGAN, StyleGAN Для задач классификации: |
| Обработка одномерных и многомерных | Архитектуры: Unet, ResNet, Pixel, Patch-GAN Безусловный синтез ProGAN, StyleGAN Для задач классификации: SampleCNN, SincNet, EnvNet, Backbone-Based Для задач устранения шума: |
| Обработка одномерных и многомерных | Архитектуры: Unet, ResNet, Pixel, Patch-GAN Безусловный синтез ProGAN, StyleGAN Для задач классификации: SampleCNN, SincNet, EnvNet, Backbone-Based Для задач устранения шума: Wave-U-Net, Demucs |
| Обработка одномерных и многомерных | Архитектуры: Unet, ResNet, Pixel, Patch-GAN Безусловный синтез ProGAN, StyleGAN Для задач классификации: SampleCNN, SincNet, EnvNet, Backbone-Based Для задач устранения шума: Wave-U-Net, Demucs Для задач регрессии: |
| Обработка одномерных и многомерных | Архитектуры: Unet, ResNet, Pixel, Patch-GAN Безусловный синтез ProGAN, StyleGAN Для задач классификации: SampleCNN, SincNet, EnvNet, Backbone-Based Для задач устранения шума: Wave-U-Net, Demucs |

^{*} Доступно автоматизированное портирование для выполнения (inference) на отечественных вычислителях

Ниже представлены решения, занимающие на данный момент высокорейтинговые позиции в различных бенчмарках (качество работы на публичной базе данных). Данные решения «интегрированы» в «Платформу ГНС» и доступны для тестирования и сравнения результатов с готовыми и пользовательскими решениями.

| Для задачи классификации изображений | Для задачи обнаружения объектов |
|--|---|
| Global Filter Networks for Image Classification ViT-H/14 | Focal Modulation Networks Hierarchical Shot Detector |

Также пользователь, используя свои данные, может провести обучение и тестирование с помощью популярных решений «интегрированных» из GitHub. Для задачи обнаружения объектов доступны следующие архитектуры:

- DDOD
- TOOD
- Deformable DETR
- YOLOX
- YOLOF
- Sparse R-CNN
- VarifocalNet
- DETR
- Dynamic R-CNN
- Side-Aware Bound
 - ary Localization

- AutoAssign
- ATSS
- Double-Head
 - R-CNN
- Foveabox
- FreeAnchor
- RepPoints
- FCOS
- TridentNet
- Libra R-CNN
- CenterNet

- FSAF
- Guided Anchoring
- Grid R-CNN
- YOLOv3
- Cascade R-CNN
- RetinaNet
- RPN
- NAS-FCOS

В «Платформе-ГНС» «интегрированные» решения запускаются в тестовом режиме, без доведения до аппаратной реализации.

Фреймворки машинного обучения и библиотека PlatLib

При разработке конечных решений «Платформа-ГНС» поддерживает возможность использования всех основных известных фреймворков машинного обучения: PyTorch, TensorFlow, Keras, Caffe, и др. Также поддерживается экспорт нейросетевых моделей в формате ONNX во встроенных решениях «Платформы-ГНС».

В качестве базового фреймворка машинного обучения в типовых «Платформы-ГНС» используются либо PvTorch, «ГосНИИАС» разработанный В ФАУ полностью оригинальный отечественный фреймворк PlatLib. Внутреннее ядро библиотеки Plat-Lib реализовано на языке C++, что позволило обеспечить эффективную реализацию процедур обучения нейросетей на различных аппаратных платформах, включая отечественные. При этом внешний программный интерфейс PlatLib реализованный на языке Python по своей структуре полностью унифицирован с наиболее популярным зарубежным фреймворком PyTorch. Это позволяет максимально быстро переносить решения с PyTorch на PlatLib и обратно. Кроме того, специалистам, имеющим опыт разработки на РуТоrch, не требуется никакого дополнительного обучения для перехода на PlatLib.

Особенности PlatLib:

- Использование динамических графов при реализации процесса обучения;
- Поддержка распределенного обучения средствами MPI, в перспективе NCCL, RCCL;
- Поддержка обучения на вычислителях производства NVIDIA, AMD, Elbrus;
- Поддержка работы на процессорах x86-64, NVIDIA, Elbrus;
- Возможность сохранения и загрузки моделей в формате «Платформы-ГНС» (обеспечивает автоматизированное портирование обученных моделей на различные аппаратные платформы встроенными средствами «Платформы-ГНС»).



Контакты

support.plat@gosniias.ru 125167, г. Москва, ул. Викторенко, 7