

3 Автоматизированное управление персоналом: нужна ли компании HRM-система? [Электронный ресурс]. – 2024. – Режим доступа: <https://hr-portal.ru/article/avtomatizirovannoe-upravlenie-personalom-nuzhna-li-kompanii-hrm-sistema>. – Дата доступа: 02.04.2024.

УДК 004.8:004.4

А. Г. Батура

НЕЙРОСЕТИ ДЛЯ НАПИСАНИЯ КОДА

В статье исследуются нейросети, которые становятся популярным инструментом в области написания кода, предлагая новые подходы и возможности для программистов. Приводятся обзор и анализ современных методов и результатов, связанных с использованием нейросетей в процессе написания кода. Рассматриваются различные архитектуры нейросетей, применяемые для автодополнения кода, генерации кода по описанию, исправления ошибок, генерации тестовых случаев и оптимизации кода.

В последние годы нейросети, основанные на глубоком обучении, стали привлекать все больше внимания в области разработки программного обеспечения. Одной из наиболее захватывающих и перспективных областей применения нейросетей является генерация кода. Нейросети для написания кода обладают потенциалом упрощения и автоматизации процесса разработки, предлагая новые инструменты и подходы для программистов.

Одной из ключевых особенностей нейросетей для написания кода является их способность анализировать контекст и синтаксическую структуру, чтобы генерировать синтаксически верный код. Они способны учитывать типы переменных, операторы и правила языка программирования, что позволяет создавать код, соответствующий синтаксическим требованиям.

Более того, некоторые нейросети основаны на предобученных моделях, таких как GPT и BERT, которые обучаются на огромном количестве текста, включая код. Это позволяет им генерировать код на различных языках программирования и преобразовывать описания функциональности в рабочие кодовые фрагменты. Благодаря этому разработчики могут сосредоточиться на более сложных задачах, в то время как нейросети выполняют рутинные и повторяющиеся задачи.

Нейросети для написания кода представляют собой быстроразвивающуюся область исследований, которая может значительно изменить способ, которым разрабатывается программное обеспечение. Далее исследуются возможности, которые реализуют нейросети для написания кода.

С развитием технологий глубокого обучения основные возможности нейросетей для написания кода включают следующие аспекты:

1 Обучение на больших объемах кода. Нейросети для написания кода требуют обширных наборов данных, чтобы научиться моделировать синтаксические и семантические правила языков программирования. Они обучаются на огромных объемах открытого и закрытого исходного кода, включая различные языки программирования и различные проекты.

2 Моделирование контекста. Нейросети для написания кода стремятся понять синтаксические и семантические связи между различными фрагментами кода. Они учитывают контекст, чтобы генерировать синтаксически верные и логически консистентные кодовые фрагменты. Это позволяет учиться извлекать информацию из предыдущего кода и использовать ее для генерации нового кода.

3 Архитектура нейросетей. При написании кода применяются различные архитектуры нейронных сетей, такие как рекуррентные нейронные сети (RNN), сверточные нейронные сети (CNN) и трансформеры. Рекуррентные нейронные сети хорошо подходят

для работы с последовательностями кода, сверточные нейронные сети могут быть использованы для анализа структуры кода, а трансформеры обладают способностью моделировать долгосрочные зависимости в коде.

4 Генерация и дополнение кода. Нейросети для написания кода могут генерировать новый код на основе входных данных, например, по описанию функциональности или тестовым примерам. Они также могут использоваться для дополнения существующего кода, предлагая автозаполнение или исправление ошибок. Это помогает программистам повысить производительность и точность своей работы.

5 Оценка и оптимизация генерируемого кода. Важной задачей является оценка качества и правильности генерируемого кода. Нейросети для написания кода могут использовать метрики, такие как синтаксическая корректность, семантическая согласованность и выполнение заданных спецификаций. Они также могут применяться для оптимизации кода, улучшая его производительность или читаемость.

6 Взаимодействие с программистами. Нейросети для написания кода могут быть интегрированы в различные инструменты разработки, такие как интегрированные среды разработки (IDE) или расширения браузера. Они могут предлагать подсказки, автодополнение и исправление ошибок, чтобы помочь программистам ускорить процесс разработки.

Однако важно отметить, что нейросети не могут заменить программиста полностью, а скорее служат инструментом для помощи и повышения производительности в процессе разработки программного обеспечения.

Рассмотрим несколько примеров применения нейросетей для написания кода и проанализируем их преимущества и недостатки (таблица 1) [1].

Таблица 1 – Примеры нейросетей для написания кода

Название нейросети	Сайт нейросети	Описание
1	2	3
COPILOT	https://github.com/features/copilot/	<p>Copilot – это нейросетевая модель, разработанная компанией OpenAI. Она является продуктом совместной работы между человеком и искусственным интеллектом, предназначенным для поддержки программистов в процессе разработки программного кода.</p> <p>Преимущества:</p> <ul style="list-style-type: none"> – использует репозиторий GitHub; – доступна с такими редакторами, как Visual Studio, JetBrains, Neovim и другими; – основана на модели Codex от OpenAI. <p>Недостаток: после окончания бесплатной пробной версии требуется покупка.</p>
TABNINE	tabnine.com	<p>Tabnine – это интеллектуальный автодополнитель кода, который использует нейронные сети для предложения контекстно-зависимых вариантов кода во время разработки программного обеспечения.</p> <p>Преимущества:</p> <ul style="list-style-type: none"> – локальная работа на сервере или в облаке; – дописывает код, изучая стиль кодирования программиста; – работает на большинстве популярных языков программирования. <p>Недостатки:</p> <ul style="list-style-type: none"> – бесплатный план не может писать большие блоки; – дорогостоящий Pro-пакет.

Окончание таблицы 1

1	2	3
FIG	fig.io	<p>Эта нейросеть создана, чтобы ускорить написание кода и сократить количество ручного ввода. Fig позволяет автозаполнять строки и проверять код на ошибки. Также инструмент позволяет безопасно синхронизировать ключи API, порты и другие конфигурации приложений между разработчиками и совместно использовать учетные данные SSH.</p> <p>Преимущество: удобное расширение для автозаполнения и исправления ошибок.</p> <p>Недостаток: пока доступна лишь MacOS версия.</p>
ASKCODI	askcodi.com	<p>AskCodi – это нейросеть, построенная на основе OpenAI Codex. Она помогает разработчикам получать максимально быстро точные ответы на их вопросы из области программирования. Среди основных функциональных возможностей можно выделить:</p> <ul style="list-style-type: none"> – генерация кода программных продуктов; – предоставление объяснений по кодам в ответ на запрос разработчика; – генерирование юнит-тестов; – объяснение документации. <p>Преимущество: бесплатная версия мало отличается от платной.</p> <p>Недостаток: модель долго учится понимать запросы. Требуется усидчивый промпт-инжиниринг.</p>
ADRENA LINE	useadrenaline.com	<p>Adrenaline – сервис для помощи разработчикам в поиске ошибок программного кода. Adrenaline – это нейросеть, способная проверять присланный код, проводить его аудит и давать рекомендации по исправлению ошибок. Сервис значительно экономит время разработчика на поиск и исправления неточностей в приложениях и сайтах.</p> <p>Преимущества:</p> <ul style="list-style-type: none"> – бесплатна и не требует регистрации. – открытый исходный код на GitHub. <p>Недостаток: специализирована только на задачах отладки и объяснения кода.</p>

Получение точного ответа от нейросетей при создании кода может быть сложной задачей, поскольку нейросети, такие как генеративно-сопоставительные сети (GAN) или языковые модели, обычно предлагают вероятностные предсказания или генерируют различные варианты ответов.

Однако существуют некоторые стратегии, которые могут помочь повысить точность и надежность ответов нейросетей при создании кода:

1 Точная постановка задачи. Чем более точно и ясно определяется задача или требования к коду, тем более вероятно, что нейросеть сможет предложить соответствующий ответ. Следует четко формулировать ожидаемый результат, типы данных, форматы и другие особенности кода, который необходимо сгенерировать.

2 Качественные данные обучения. Обучение нейросети на качественных данных является ключевым фактором для получения точных ответов. Необходимо предоставить модели обширных и разнообразных данных, содержащих высококачественные примеры кода, которые соответствуют определенной задаче.

3 Подбор правильной архитектуры модели. Выбор правильной архитектуры модели может существенно повлиять на точность ответов. Различные типы нейросетей имеют свои преимущества и ограничения, и выбор подходящей архитектуры может быть важным шагом для получения точных результатов.

4 Тщательная настройка гиперпараметров. Гиперпараметры модели, такие как размер эмбедингов, количество слоев, скорость обучения и другие, могут влиять на точность ответов. Процесс настройки гиперпараметров требует экспериментирования и итеративного подбора оптимальных значений.

5 Постобработка и проверка. После получения ответа от нейросети, рекомендуется провести постобработку и проверку сгенерированного кода. Это может включать статический анализ, тестирование и рецензирование кода для проверки его правильности, безопасности и соответствия требованиям проекта.

Важно отметить, что хотя нейросети могут быть мощными инструментами для генерации кода, они не являются идеальными и могут допускать ошибки или предлагать неправильные решения. Поэтому всегда рекомендуется использовать здравый смысл, проверять и рецензировать сгенерированный код и полагаться на экспертные знания разработчика при принятии окончательных решений.

Литература

1 15 нейросетей для написания кода в 2023 году [Электронный ресурс]. – 2023. – Режим доступа: <https://wotpack.ru/15-nejrosetej-dlja-napisanija-koda-v-2023-godu/> ?ysclid=lp6m97qfzd155792539. – Дата доступа: 08.12.2023.

УДК 004.383.5

В. В. Батура

ПРИМЕНЕНИЕ НЕЙРОСЕТЕЙ В ДИЗАЙНЕ ВЕБ-САЙТОВ

В статье исследуются возможности применения нейросетей в дизайне веб-сайтов, рассматриваются онлайн-сервисы Figma и TurboLogo для этих целей, анализируются их преимущества и недостатки. Результаты исследования показывают перспективы развития данной технологии и позволяют сделать вывод, что использование нейросетей в дизайне не заменяет человеческого творческого мышления.

На сегодняшний день нейросети являются одним из самых перспективных направлений искусственного интеллекта, которое находит широкое применение в различных областях. В последние годы они также начали проникать в сферу дизайна веб-сайтов, предлагая пользователям новые возможности [1].

Нейросети применяются в веб-дизайне для различных целей, которые помогают улучшить процесс создания и оптимизации веб-сайтов.

Нейронная сеть может анализировать, как пользователи реагируют на различные элементы интерфейса сайта, насколько пользователю удобно ими пользоваться и предлагать улучшения для более привлекательного дизайна. Также искусственный интеллект может помогать в разработке макета сайта.