УДК 004.852

СТРУКТУРИРОВАНИЕ КОНТЕКСТА ЯЗЫКОВЫХ МОДЕЛЕЙ В ЗАДАЧАХ ПОДДЕРЖКИ ПРОЦЕССОВ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С УЧЕТОМ ОСОБЕННОСТЕЙ ПОВЕДЕНИЯ И ПРОФИЛЕЙ РАЗРАБОТЧИКОВ

Витко М.А. (Университет ИТМО)
Научный руководитель – кандидат технических наук, Ковальчук С.В.
(Университет ИТМО)

Введение. С течением времени продукты программного обеспечения становятся всё более сложными и многофункциональными, что подталкивает разработчиков к поиску новшеств и формированию инновационных подходов к разработке программного обеспечения. Разработчики, в свою очередь, отличаются друг от друга стилем работы, уровнем знаний и наработанным опытом. Именно поэтому подход, учитывающий индивидуальность каждого в использовании вспомогательных инструментов, может кардинально усилить их эффективность.

Недавние достижения в области языковых моделей, представленные такими моделями, как GPT и BERT, открывают возможности для автоматизации и улучшения различных аспектов процесса разработки программного обеспечения [1].

Данное исследование нацелено на создание методики, которая позволит формировать контекст запроса к языковым моделям так, чтобы они учитывали индивидуальные черты и предпочтения разработчиков, что позволит повысить эффективность взаимодействия «разработчик ПО - языковая модель».

Основная часть. Разработанная структура контекста взаимодействия охватывает параметры, характеризующие профиль разработчика:

1) Опыт работы (начинающий, опытный, эксперт).
2) Тип взаимодействия (написание кода, тестирование, процесс обучения).
3) Конкретные технические детали (используемый язык программирования, используемые библиотеки, используемые API).

Помимо деталей, указывающих на профиль разработчика программного обеспечения, структура контекста учитывает критерии, необходимые для составления эффективного запроса в общем случае взаимодействия [2]: запрос должен быть составлен чётко и конкретно; необходимо описание контекста возникшей проблемы, указание условий решения; в запросе должен содержаться вид результата или тип желаемого решения; должны быть указаны предыдущие попытки и известные ограничения (если такие имеются).

Для оценки эффективности разработанной структуры контекста взаимодействия «разработчик программного обеспечения - языковая модель», использованы следующие метрики: Human Evaluation, BLEU, CodeBLEU.

Использование метрики Human Evaluation позволило оценить адаптируемость языковой модели к более насыщенному деталями контексту с точки зрения человеческого восприятия [3]. Запросы формировались согласно различным сценариям взаимодействия разработчика с языковой моделью с учетом профиля разработчика. Метрика показала, что по мере насыщения контекста деталями, учитывающими профиль разработчика, языковая модель (GPT-3.5) формирует ответ, способный помочь решить рассматриваемую задачу с наибольшей эффективностью.

Для проведения эксперимента с целью оценки по метрикам BLEU и CodeBLEU был использован набор данных CMU CoNaLa. Набор данных представлен 2379 обучающими и 500 тестовыми корпусами, которые были аннотированы вручную [4]. Используя регулярные выражения, был экстрагирован новый поднабор данных из первичного массива информации.

В качестве референсного текста был рассмотрен ответ с портала *StackOverflow*, который был отмечен как «The question owner accepted this as the best».

Выбор референса основан на критерии наивысшей практической полезности и смысловой значимости ответа с точки зрения пользователя, задавшего вопрос. Этот подход позволяет более объективно оценивать сгенерированные ответы в контексте их полезности и соответствия ожиданиям пользователей на основе опыта вопроса и принятого решения.

Из сгенерированного с помощью регулярных выражений корпуса был выбран словарь с уникальным идентификатором вопроса № 16868457. Для сопоставления референсному выражению используется сформированный набор ответов-кандидатов, представляющий собой ответы языковой модели (GPT-3.5) в соответствии с насыщением контекста запроса. Словари-кандидаты формировались согласно описанным составляющим структуры контекста запроса и представлены пятью контекстами, где каждый последующий включает в себя детали, указанные в предыдущем, а также информацию, представленную новыми деталями.

По результатам эксперимента значение метрики CodeBLEU по мере насыщения контекста изменяется от значения 0.16 для первого словаря-кандидата, включающего в себя исключительно информацию из заголовка вопроса № 16868457, до 0.35 для четвертого словаря-кандидата, включающего в себя информацию из заголовка, технические детали, желаемый формат ответа и информацию об опыте разработчика программного обеспечения. Для словаря-кандидата №5, дополняющего предыдущий информацией о контексте, в котором происходит разработка, значение метрики CodeBLEU снизилось до 0.31.

Согласно результатам эксперимента, по мере насыщения контекста показатели метрики BLEU изменяются в диапазоне [*8.59e-232, 1.04e-231*]. Даже в условиях очень высокой чувствительности используемой метрики к длине используемых словарей для контекста четвертого кандидата-словаря метрика показывает значения, наиболее близкие к 1.

Выводы. По результатам исследования предметной области было выполнено структурирование контекста взаимодействия «разработчик ПО – языковая модель». Проведенный эксперимент на наборе данных CoNaLa, а также эксперимент с применением метрики Human Evaluation позволили оценить эффективность разработанной структуры контекста запроса к языковой модели: по мере насыщения контекста деталями, учитывающими профиль разработчика, качество сгенерированного ответа растёт.

Наблюдения по насыщению контекста указывают на то, что добавление слишком большого объёма контекстной информации может не всегда способствовать повышению качества сгенерированного кода. В некоторых случаях, как показывает снижение показателей CodeBLEU для словаря-кандидата №4, перегрузка информацией может даже ухудшить результаты генерации кода.

Область дальнейших исследований представляет из себя проведение достаточного количества экспериментов с последующим выполнением статистического анализа, а также решение задачи оптимизации по включению рассмотренных элементов контекста в запрос.

Список использованных источников:

1. Jalil, S. (2023). “The Transformative Influence of Large Language Models on Software Development.” ArXiv, abs/2311.16429.

2. Wang, Shuai et al. “Can ChatGPT Write a Good Boolean Query for Systematic Review Literature Search?” Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (2023).

3. Schuff H, Vanderlyn L, Adel H, Vu NT. How to do human evaluation: A brief introduction to user studies in NLP. Natural Language Engineering. 2023;29(5):1199-1222.

4. CoNaLa: The Code/Natural Language Challenge: сайт. Pittsburg. URL: https://conala-corpus.github.io/ (дата обращения: 05.02.2024).