УДК 004.932:004.8

«ПОСТРОЕНИЕ СТРУКТУРНОГО ПРЕДСТАВЛЕНИЯ ВЕБ-ИНТЕРФЕЙСОВ НА ОСНОВЕ CV/ML-ПОДХОДА С ПРИМЕНЕНИЕМ МЕЖПРЕДСТАВЛЕНЧЕСКОГО СВЯЗЫВАНИЯ»

Асачьев Артём Борисович
artem662000@mail.ru
магистрант 1 курса образовательной программы «Информационные системы»
Торайгыров Университет, г. Павлодар, Республика Казахстан
Научный руководитель - кандидат педагогических наук, ассоц. профессор Найманова Д.С.

Аннотация

В работе рассматривается задача автоматического построения структурированной разметки веб-интерфейсов на основе визуальных данных. Предлагается подход, основанный на применении методов компьютерного зрения и машинного обучения для выделения элементов графического пользовательского интерфейса с последующим формированием иерархической структуры. В отличие от существующих решений, использующих мультимодальные языковые модели, такие как GPT-4V, предлагаемый метод ориентирован на построение явного представления интерфейса, пригодного для последующего использования в системах автоматизации [1]. Показано, что предложенный подход обеспечивает повышение скорости обработки изображений и позволяет эффективно работать с интерфейсами большой сложности. Выявлены ограничения, связанные с детекцией мелких объектов, и предложено решение на основе многоуровневого анализа.

Введение. Современные информационные системы характеризуются высокой сложностью графических пользовательских интерфейсов. Такие интерфейсы содержат большое количество визуальных элементов, взаимосвязанных как на уровне представления, так и на уровне бизнес-логики. Автоматизация взаимодействия с подобными системами требует не только распознавания отдельных элементов, но и понимания их структуры и роли в рамках пользовательских сценариев.

Ранее в работе автора была предложена концепция построения графа бизнес-процессов (BPG, Business Process Graph - граф бизнес-процессов), в котором элементы интерфейса, их представления и действия связываются в единую структуру. Ключевым элементом данной концепции является механизм межпредставленческого связывания (cross-view linking - связывание представлений одной сущности в разных интерфейсах), позволяющий объединять различные визуальные проявления одной и той же сущности.

С развитием мультимодальных моделей, таких как GPT-4 Turbo и GPT-4V, появилась возможность интерпретировать интерфейсы напрямую на основе изображений. Однако данные подходы не обеспечивают устойчивого построения структурированной модели интерфейса и демонстрируют ограничения при масштабировании [1, 2].

Целью данной работы является разработка метода автоматической разметки интерфейсов, который формирует явное структурное представление и может использоваться как основа для построения BPG.

Анализ существующих подходов. Существующие методы анализа пользовательских интерфейсов можно условно разделить на два класса: методы компьютерного зрения и мультимодальные языковые модели.

Методы компьютерного зрения традиционно используются для детекции объектов на изображениях [3]. В контексте интерфейсов такие методы позволяют выделять кнопки, текстовые поля, списки и другие элементы. В работах, посвящённых применению глубокого обучения, используются сверточные нейронные сети и трансформеры для

извлечения визуальных признаков и локализации объектов [4]. Дополнительно применяется оптическое распознавание символов (OCR, Optical Character Recognition - оптическое распознавание текста), что позволяет извлекать текстовую информацию из интерфейса [3].

Основным преимуществом данного подхода является высокая скорость обработки и возможность масштабирования на большие объёмы данных. Однако такие методы ограничены в способности интерпретировать семантику элементов и их роль в пользовательских сценариях.

Альтернативный подход основан на использовании мультимодальных языковых моделей, таких как GPT-4V и Claude 3 [1, 5]. Эти модели способны анализировать изображение и генерировать текстовые описания, включая последовательности действий [6]. В ряде исследований показано, что подобные модели могут использоваться в качестве агентов для взаимодействия с интерфейсами [2].

Тем не менее, использование LLM связано с рядом ограничений. Во-первых, высокая вычислительная стоимость делает их применение затруднительным в задачах массовой обработки. Во-вторых, отсутствие явной структуры приводит к нестабильности результатов. В-третьих, такие модели не формируют долговременное представление интерфейса, что ограничивает их применение в сложных сценариях автоматизации.

Предлагаемый подход. Предлагаемый метод направлен на устранение указанных ограничений за счёт явного построения структуры интерфейса. Основу метода составляет конвейер обработки, включающий этапы детекции элементов, извлечения признаков, кластеризации и построения иерархии.

На первом этапе осуществляется детекция элементов интерфейса. Для этого используются модели компьютерного зрения, позволяющие выделять базовые компоненты, такие как кнопки, текстовые блоки и контейнеры. Параллельно выполняется OCR для извлечения текстового содержимого.

На втором этапе формируется представление каждого элемента. Для этого вычисляются визуальные эмбеддинги с использованием моделей, подобных CLIP [7], а также текстовые эмбеддинги на основе трансформеров. Дополнительно учитываются пространственные характеристики, включая координаты и размеры элементов.

Следующим этапом является кластеризация, в рамках которой элементы, принадлежащие одной сущности, объединяются в группы. Данный процесс реализует механизм cross-view linking (межпредставленческое связывание), описанный в предыдущей работе. Кластеризация выполняется на основе многомодального сходства, включающего визуальные, текстовые и пространственные признаки.

После этого строится иерархическая структура интерфейса. В работе были рассмотрены два альтернативных подхода: top-down (tree-to-leaf - от дерева к листьям) и bottom-up (leaf-to-tree - от листьев к дереву).

В подходе tree-to-leaf сначала формируется глобальная структура страницы, после чего она декомпозируется на более мелкие элементы. Данный метод предполагает наличие устойчивой структуры интерфейса и чувствителен к ошибкам на верхнем уровне.

В подходе leaf-to-tree, напротив, сначала детектируются атомарные элементы, после чего они агрегируются в более крупные структуры. Такой подход менее чувствителен к шуму и лучше адаптируется к реальным интерфейсам, которые часто не имеют строгой иерархии.

В ходе экспериментов было установлено, что подход leaf-to-tree обеспечивает более устойчивые результаты, что обусловило его выбор в качестве основного.

Архитектурно решение организовано в виде модульной системы, включающей слой обработки изображений, слой построения графа (BPG) и слой интеграции с языковыми моделями. Такое разделение позволяет использовать LLM исключительно для задач интерпретации и планирования, сохраняя при этом контроль над структурой данных.

Одним из ключевых аспектов предложенного метода является формирование устойчивого представления элементов интерфейса, независимого от конкретного визуального контекста. В традиционных подходах элементы рассматриваются в рамках одного изображения, что ограничивает возможность обобщения. В данной работе используется механизм межпредставленческого связывания (cross-view linking - связывание различных визуальных представлений одной сущности), позволяющий объединять элементы, обнаруженные на различных экранах или состояниях интерфейса.

Для этого вводится метрика сходства, учитывающая несколько компонент. Визуальная составляющая определяется на основе эмбеддингов, полученных с использованием моделей, аналогичных CLIP. Текстовая составляющая формируется на основе результатов OCR с последующим преобразованием в векторное пространство. Пространственная составляющая учитывает относительное расположение элементов. Дополнительно используется временной контекст, извлекаемый из последовательностей пользовательских действий.

Комбинация указанных факторов позволяет существенно повысить точность кластеризации. В частности, элементы, имеющие различные визуальные представления, но совпадающие по функциональной роли, могут быть корректно объединены в одну сущность. Это является важным шагом для последующего построения графа бизнес-процессов (BPG, Business Process Graph - граф бизнес-процессов).

Следует отметить, что использование кластеризации требует балансировки между точностью и устойчивостью. Слишком агрессивное объединение приводит к потере различий между элементами, тогда как избыточная фрагментация усложняет последующую обработку. В работе применяется эмпирически подобранная комбинация методов плотностной кластеризации и пороговых правил, что позволяет достичь приемлемого компромисса.

Эксперименты и результаты. Экспериментальная оценка показала, что предложенный подход обладает высокой производительностью при обработке интерфейсов. В сравнении с методами, основанными на использовании GPT-4V, достигается существенное снижение времени обработки одного изображения. Это связано с тем, что CV-модели работают локально и не требуют сложных генеративных вычислений.

Кроме того, продемонстрирована способность метода работать с интерфейсами большой сложности, содержащими сотни элементов. В таких условиях использование LLM оказывается затруднительным из-за ограничений контекста.

В процессе экспериментов была выявлена проблема, связанная с детекцией мелких объектов, расположенных на изображениях большого размера. При глобальной обработке такие элементы теряются из-за недостаточного разрешения признаков.

Для решения данной проблемы был предложен механизм многоуровневого анализа. Изображение разбивается на более мелкие области, каждая из которых обрабатывается отдельно. Затем результаты объединяются в единую структуру. Данный подход позволил существенно повысить точность детекции мелких элементов.

Дополнительный анализ производительности показал, что предложенный подход демонстрирует линейную зависимость времени обработки от количества элементов интерфейса. Это особенно важно при работе с промышленными системами, содержащими сложные формы, таблицы и вложенные компоненты.

В отличие от этого, использование мультимодальных моделей приводит к росту вычислительных затрат, обусловленному увеличением размера входного контекста. При обработке больших интерфейсов возникает необходимость разбиения изображения на части, что влечёт за собой дополнительные накладные расходы и усложняет агрегацию результатов.

Отдельное внимание было уделено устойчивости метода к изменениям интерфейса. В ходе экспериментов моделировались сценарии, в которых изменялись стили, размеры элементов и их расположение. Было показано, что использование многомодальных

признаков позволяет сохранять корректность кластеризации даже при значительных визуальных изменениях.

Также была проведена оценка качества детекции мелких элементов до и после внедрения многоуровневого анализа. В базовой версии алгоритма доля корректно распознанных мелких элементов была существенно ниже по сравнению с крупными объектами. После внедрения разбиения изображения на локальные области данный показатель значительно улучшился, что подтверждает эффективность предложенного решения.

Обсуждение. Полученные результаты показывают, что использование явной структуры интерфейса позволяет преодолеть ограничения, характерные для мультимодальных языковых моделей. В частности, достигается повышение предсказуемости и интерпретируемости системы.

При этом следует отметить, что LLM сохраняют свою ценность в задачах, связанных с интерпретацией и генерацией действий. Наиболее перспективным направлением является комбинирование предложенного подхода с моделями типа GPT-4V, где структурированное представление используется в качестве контекста.

Дополнительным преимуществом предложенного подхода является возможность интеграции с системами автоматизации, использующими явные правила и ограничения. В отличие от LLM, которые генерируют действия на основе вероятностных оценок, структурированное представление позволяет выполнять валидацию действий до их исполнения.

Это особенно важно в задачах, связанных с бизнес-процессами, где ошибки могут приводить к некорректным операциям. Использование BPG позволяет явно задавать предусловия (preconditions - предусловия выполнения действий) и постусловия (postconditions - ожидаемые результаты), что делает систему более надёжной.

Кроме того, предложенный подход обеспечивает лучшую интерпретируемость. Каждое действие может быть объяснено через структуру графа, что упрощает отладку и анализ системы. В случае использования LLM подобная интерпретация зачастую отсутствует или носит постфактум характер.

При этом следует подчеркнуть, что мультимодальные модели не рассматриваются как конкурирующее решение, а скорее как дополнительный компонент. Их использование целесообразно на этапе интерпретации пользовательских намерений и генерации высокоуровневых планов действий.

Заключение.В работе предложен метод автоматической разметки пользовательских интерфейсов на основе компьютерного зрения и машинного обучения. Разработан конвейер обработки, включающий детекцию элементов, извлечение признаков, кластеризацию и построение иерархической структуры.

Проведено сравнение с современными мультимодальными моделями, показавшее преимущества предложенного подхода с точки зрения производительности и масштабируемости. Выявлены ограничения и предложены методы их устранения.

Результаты работы могут быть использованы в дальнейшем для построения систем автоматизации, основанных на графах бизнес-процессов.

Список использованной литературы:

1. OpenAI. GPT-4V System Card. — 2023. — 25 с.
2. OpenAI. GPT-4 Technical Report // arXiv:2303.08774. — 2023. — 94 с.
3. Liao M., Shi B., Bai X. et al. TextBoxes++: A Single-Shot Oriented Scene Text Detector // IEEE Transactions on Image Processing. — 2018. — Vol. 27. — 3676–3690 с.
4. Vaswani A., Shazeer N., Parmar N. et al. Attention Is All You Need // Advances in Neural Information Processing Systems. — 2017. — 5998–6008 с.
5. Anthropic. Claude 3 Model Card. — 2024. — 18 с.

6. Xu K., Ba J., Kiros R. et al. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention // Proceedings of ICML. — 2015. — 2048–2057 с.
7. Radford A., Kim J. W., Hallacy C. et al. Learning Transferable Visual Models From Natural Language Supervision // Proceedings of ICML. — 2021. — 8748–8763 с.