ISSN 2658-5014 (Print) СИИТ ISSN 2686-7044 (Online)
http://siit.ugatu.su 2026. Т. 8, № 1(25)
СИСТЕМНАЯ ИНЖЕНЕРИЯ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
Научная статья

УДК 004.89 DOI: 10.54708/SIIT-2026-no1-p101 EDN: JRFRIP

Мультиагентная система для решения задачи мультимодального преобразования видеолекции в текстовый документ

М. Е. Исмагулов

Югорский государственный университет

В статье рассматривается мультиагентная система, разработанная для автоматизированного преобразования видеолекций в полноценный текстовый конспект, который отражает содержание исходного видеофайла с учетом информации из аудио- и видеомодальностей. Целью исследования является создание решения для автоматической трансформации образовательного видеоконтента в структурированный текст, пригодный для дальнейшего использования в учебных целях. В работе применяются методы машинного обучения, включая глубокие нейронные сети, а также подходы мультиагентных систем для координации сложных процессов обработки данных. Разработан прототип системы, реализованный на основе архитектуры «оркестратор-исполнитель». Данная архитектура включает три типа агентов: оркестратор, отвечающий за управление и координацию процессов, агент исполнитель, использующий интеллектуальные модели машинного обучения для анализа контента и агент-инструмент, выполняющий детерминированную обработку данных. На текущем этапе прототип способен обрабатывать один из трех запланированных форматов видеолекций, создавая текстовый документ в формате Markdown. Для обучения и тестирования системы сформирован набор данных, основанный на реальных записях, онлайн-курсов. Метрики прототипа включают: для модели OpenAI Whisper Medium — WER 16,3%; для конвейера YOLO11-PySceneDetect-pHash — Precision: 0.94, Recall: 1.00, F1-Score: 0.97; точность оптического распознавания символов составила 94,89% по результатам бенчмарка. Особенность мультиагентной системы заключается в использовании последовательных конвейеров обработки данных, объединяющих несколько алгоритмов и моделей. В заключение представлены результаты текущего этапа разработки, а также намечены направления дальнейшего совершенствования системы, такие как расширение поддержки форматов и улучшение производительности

Мультимодальная обработка видеолекции; мультиагентные системы; паттерн оркестратор-исполнитель; конвейерная обработка данных; прототип мультиагентной системы.

Введение

Задача автоматического преобразования видеолекций в текстовые документы с использованием мультимодальных подходов актуальна для различных образовательных процессов. Например, такие системы позволят платформам массовых онлайн-курсов автоматически создавать текстовые версии загружаемых видеолекций [Gon23]. Кроме того, они дадут возможность получать текстовые документы на основе записей очных занятий [Tak24]. Еще одно применение таких систем — интеграция в платформы видеоконференций для автоматического создания протоколов или текстовых отчетов по итогам вебинаров и онлайн-конференций.

На момент исследования актуальными методами решения задач мультимодального преобразования образовательного видеоконтента являются методы, основанные на использовании

Рекомендовано к публикации программным комитетом XI Международной научной конференции ITIDS'2025 «Информационные технологии интеллектуальной поддержки принятия решений», Уфа, 13–15 ноября 2025 г.

Исмагулов М. Е. Мультиагентная система для решения задачи мультимодального преобразования видеолекции в текстовый документ // СИИТ. 2026. Т. 8, № 1(25). С. 101-110. DOI: 10.54708/SIIT-2026-no1-p101. EDN: JRFRIP.

Ismagulov M. E. "Multi-agent system for solving the problem of multimodal video lecture-to-text transformation" // SIIT. 2026. Vol. 8, no. 1(25), pp. 101-110. DOI: 10.54708/SIIT-2026-no1-p101. EDN: JRFRIP. (In Russian).

мультимодальных больших языковых моделей [Wan23, Ata24, Luo20]. Они позволяют анализировать все модальности, одновременно извлекая более полный контекст, также подобный метод позволяет учитывать связи между модальностями [Ye23].

Однако помимо преимуществ данные методы также имеют свои недостатки:

- Многие модели не обеспечивают учет всех ключевых этапов, отраженных в видеолекциях, следовательно, страдает качество выходного материала и потеря понятий, отраженных в видеолекциях [Zou24];
- Сложность алгоритмической реализации подобных методов, мультимодальные большие языковые модели сложны для обучения [Wu23];
- Высокие требования к обучающему набору данных, как к структуре данных, так и к их объему [Li24];
- Высокие требования к вычислительным ресурсам для обучения подобных моделей и для их эксплуатации [Xu24].

В данном исследовании предлагается альтернативный метод решения подобных задач, а именно метод, основанный на мультиагентных системах. Идея заключается в следующем: видеолекцию как мультимодальный объект необходимо декомпозировать на отдельные модальности, далее полученные модальности анализировать отдельными агентами в составе алгоритмов которых, будут модели машинного обучения, заточенные на решение узкоспециализированных задач.

После анализа агентами на выходе для каждой модальности появятся текстовые файлы, отражающие содержимое этих модальностей, на последнем этапе тексты агрегируются согласно ключевой характеристике – времени. Агент, агрегирующий результаты должен обладать способностью удалять артефакты от других моделей такие как: ошибки текста, его дублирование и т. д. Также агенту агрегирующий тексты необходимо задавать структуру документа и вставлять изображения из сцен видеолекции.

Описание мультиагентной системы

Мультиагентная система — это распределенная система, состоящая из множества взаимодействующих автономных программных агентов. Коллективное поведение этих агентов позволяет решать сложные задачи, не поддающиеся решению централизованными методами [Dor18].

Одним из видов мультиагентной системы является «Оркестратор-Исполнитель». В данной архитектуре агенты делятся на 3 вида:

1. Агент оркестратор — это специализированный вид агента чья задача координировать действия других агентов, декомпозировать сложную задачу на подзадачи, распределять их между агентами и агрегировать результат [Fou24];
2. Агент исполнитель — это вид интеллектуального агента, основанного на алгоритмах машинного обучения, имеющий алгоритмы адаптации подобные генетическим алгоритмам, задача которого выполнять подзадачи, полученные от оркестратора и передавать результаты обработки обратно оркестратору [Yan23];
3. Агент инструмент родственный агенту-исполнителю, но отличается тем, что не имеет в своей основе алгоритмы машинного обучения или механизмы адаптации, а основан на детерминированных алгоритмах, функция которого это выполнение операций по обработке данных [Mil25].

Формализация мультиагентной системы

Схема взаимодействия оркестратора и агентов-исполнителей может быть формализована средствами теории графов. На рис. 1 представлен ориентированный граф, описывающий структуру взаимодействия компонентов.

В данной схеме вершина O соответствует оркестратору, вершины W_1, …, W_i обозначают агентов-исполнителей, при этом индекс i отражает возможность масштабирования числа агентов в составе мультиагентной системы. Рёбра графа Q интерпретируются как сообщения, передаваемые между оркестратором и агентами-исполнителями. Передача сообщений осуществляется в обоих направлениях: от оркестратора к агентам (например, в форме управляющих команд) и от агентов к оркестратору (например, в виде уведомлений о завершении обработки).

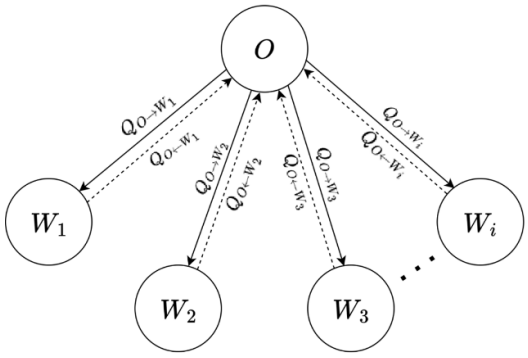

Рис. 1 Граф взаимодействия оркестратора и агентов-исполнителей

В рамках данного исследования реализован именно такой метод взаимодействия компонентов системы. В качестве архитектурного подхода выбран стиль REST, обеспечивающий унифицированное взаимодействие посредством методов HTTP POST и HTTP GET. Обмен сообщениями осуществляется в формате JSON, что позволяет обеспечить формализованность коммуникации между агентами (листинг 1).

Листинг 1

Примеры сообщений

Сообщение от оркестратора к агенту-исполнителю — JSON

```
{
 "video_path": "uploads/lecture4.mp4",
 "frame_rate": "10",
 "resolution": "1920x1080"
}
```

Сообщение от агента-исполнителя к оркестратору — JSON

```
{
   "status": "success",
   "audio_path": [
      "uploads/lecture4/audio/audio.wav"
   ],
   "frames_path": "uploads/lecture4/frames/lecture4_mp4.zip",
   "log_id": 17
}
```

В рамках данного исследования для описания внешнего воздействия на систему, включая взаимодействие пользователя и интеграцию с другими программными комплексами, вводится понятие окружения (Environment). Под окружением понимается среда, обеспечивающая передачу управляющих команд оркестратору — центральному компоненту, отвечающему за координацию агентов в мультиагентной системе (рис. 2) [Agr25]. Управляющие команды формируются в соответствии с формализованным синтаксисом и могут представляться в виде инструкций на языках программирования (например, Python, Bash), сетевых запросов (HTTP POST, Curl) либо структурированных сообщений в форматах JSON или XML.

Оркестратор, выступающий центральным компонентом мультиагентной системы, получает от окружения управляющие команды и объект обработки (видеолекцию). Его основная функция заключается в формировании плана обработки, декомпозиции исходной задачи на подзадачи, а также в выборе и координации агентов-исполнителей и агентов-инструментов, определённых в конфигурационном файле системы.

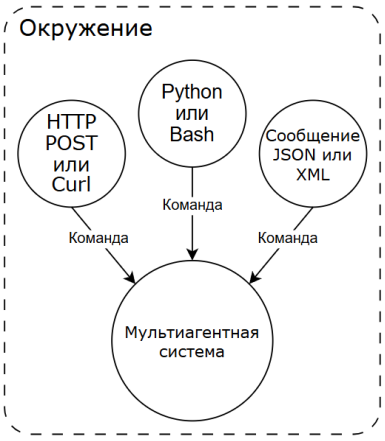

Рис. 2 Схема взаимодействия окружения и оркестратора

Рис. 3 Схема передачи данных между агентами через оркестратора

В процессе работы оркестратор обеспечивает маршрутизацию и передачу данных между агентами (рис. 3). По завершении выполнения всех подзадач он осуществляет агрегацию результатов и формирует итоговый текстовый документ в формате Markdown, который возвращается пользователю через окружение (рис. 4).

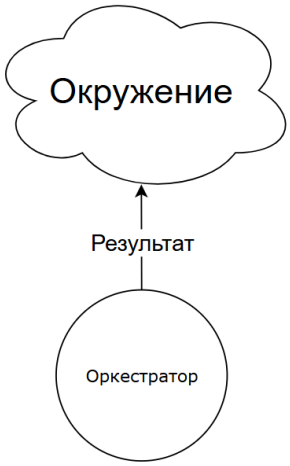

Рис. 4 Схема передачи результата от оркестратора в окружение

Для решения задачи автоматизированного преобразования видеолекции в текстовый документ в рамках данного исследования предлагается алгоритм обработки, основанный на декомпозиции исходного материала на отдельные составляющие и их анализе специализированными агентами.

Последовательность шагов включает:

1. Извлечение аудиопотока и кадров видеоряда;
2. Параллельный анализ аудиомодальности с применением модели преобразования речи в текст и видеомодальности посредством конвейера алгоритмов и моделей, направленных

на выделение сцен (сцена определяется как уникальный кадр, содержащий всю релевантную текстовую информацию на текущем временном интервале);

3. Преобразование кадров сцен в текст с использованием методов оптического распознавания символов (OCR);

4. Интеграцию текстов обеих модальностей с применением большой языковой модели, которой передаётся специализированный промпт, задающий структуру выходного документа; на данном этапе выполняется агрегация модальностей, коррекция ошибок распознавания речи и OCR, а результат передаётся оркестратору;

5. Завершение обработки и передача итогового текстового документа в окружение посредством оркестратора.

Формирование плана обработки видеолекции осуществляется оркестратором, в ядре которого реализован алгоритм на основе большой языковой модели. Данной модели передаётся промпт в формате JSON, включающий текст задания и перечень доступных агентов. Указанные элементы промпта формируются на основе конфигурационных JSON-файлов, что обеспечивает возможность реконфигурируемости мультиагентной системы посредством их изменения и подключения новых агентов. В результате языковая модель генерирует JSON-план обработки видеолекции (листинг 2). Структурная схема функционирования системы представлена на рис. 4.

Листинг 2

Пример плана

План обработки видеолекции
(ответ большой языковой модели) — JSON

```
{ "pipeline": [ {
    "agent": "frame_extractor",
    "parameters": {
      "video_path": "uploads/lecture4",
      "frame_rate": 5,
      "resolution": "1920x1080"
    } }, {
    "agent": "speech_to_text_ru",
    "parameters": {
      "audio_path": "uploads/lecture4/audio",
      "language": "ru",
      "output": "json"
    } }, {
    "agent": "frame_to_scene_converter_type_one",
    "parameters": {
      "frames_path": "uploads/lecture4/frames",
      "frame_rate": 10,
      "resolution": "1920x1080"
    } }, {
    "agent": "ocr",
    "parameters": {
      "scenes_path": "uploads/lecture4/scenes",
      "language": "ru" }
  }, {
    "agent": "llm_based_agent",
    "parameters": {
      "texts": {
        "speech_text": "uploads/lecture4/audio/audio_txt",
        "scenes_path": "uploads/lecture4/scenes",
        "ocr_text": "uploads/lecture4/frames/scenes/scenes_txt"
```

```
        },
        "model": "mistral-large-latest",
        "temperature": 0.7 }}
    ]}
```

Форматы видеолекций

В рамках исследования предполагается обработка трех форматов видеолекций, согласно этим форматам лекции планируется формировать набор данных. Основываются данные форматы на популярных видах обучающих видеороликов, выложенные на открытых видеохостингах. Форматы лекции в данном исследовании делятся на 3 типа:

- Видеолекция «лектор и сопровождающая презентация» представляет собой формат, при котором в 1/3 кадра присутствует лектор, а на оставшейся части кадра находится презентационный или иллюстративный материал, отражающий краткое содержание речи лектора в данный момент времени. Данный формат является основным при съемках в интерактивной видеостудии «Jalinga» которая создана для записи онлайн курсов;
- Формат видеолекции «презентация и закадровый сопровождающий голос», является популярным и одним из самых распространенных в видеохостингах, поскольку является очень простым не требующим сложного оборудования для видеосъемки и основан на явлении так называемого «скринкаста», записи рабочего стола компьютера посредством специальных программ;
- Формат видеолекции «лектор и маркерная или меловая доска», этот формат видеолекции является отражением явления, при котором происходит запись очной лекции.

Набор данных

Набор данных, использованный в исследовании, представляет собой записи онлайн-курсов общим объёмом около 115 ГБ, что соответствует приблизительно 27 часам видео. В качестве меток для сцен и кадров применялись CSV-таблицы (таблица).

Таблица 3

Структура CSV-таблицы меток набора данных

Id_ scene	Start_ scene	End_ scene	Path_ to_image
1	00.00.00	00.00.27	\folder\frame_1 … \folder\frame_27
2	00.00.28	00.00.51	\folder\frame_28 … \folder\frame_51
3	00.00.52	00.02.48	\folder \frame_52 … \folder\frame_168
…	…	…	…
N	hh.mm.ss	hh.mm.ss	\folder\frame_N … \folder\frame_M

В метках в качестве идентификатора сцены указывается id_scene. Временные границы сцены задаются ключами Start_scene и End_scene в формате ЧАСЫ.МИНУТЫ.СЕКУНДЫ.

Под сценой в данном исследовании понимается временной интервал видео, удовлетворяющий одному из следующих условий:

- смена ключевого контента в кадре (слайды презентации, надписи на доске и др.);
- изменение ракурса видеокамеры, включая зумирование или смещение;
- смена типа кадра, например, переход от изображения лектора к демонстрации слайда на весь экран.

В то же время, к сценам не относятся следующие изменения:

- жестикуляция лектора;
- незначительные перемещения лектора в пределах кадра;
- мимические изменения.

Результаты и обсуждение

Реализация прототипа

Прототип мультиагентной системы был реализован на языке Python в виде клиент-серверного приложения, основанного на библиотеке FastAPI. Для описания базовой модели агента использовалась библиотека Pydantic, обеспечивающая формализацию структуры агента через систему полей, задаваемых в конфигурационном файле. Пользовательский интерфейс реализован с использованием библиотеки Jinja2.

Каждый агент функционирует в изолированной среде Docker-контейнера. В качестве системы хранения данных применяется проект MinIO в связке с библиотекой Amazon S3 Storage for Python. Для хранения метаданных используется реляционная база данных PostgreSQL.

Взаимодействие оркестратора и агентов организовано через обмен сообщениями. Так, например, от оркестратора к агенту-инструменту (специализирующемуся на извлечении кадров и аудио) передаётся сообщение-команда, где указываются параметры: путь к объекту обработки (video_path), частота извлечения кадров (frame_rate) и требуемое разрешение выходных кадров. В ответ агент формирует сообщение о завершении обработки, содержащее статус выполнения (status), путь к аудиофайлу (audio_path), путь к архиву с кадрами (frames_path) и системный идентификатор задания (log_id), отражающий его позицию в логах.

Алгоритмы и модели, интегрированные в мультиагентную систему, реализованы на языках Python и C++, что характеризует разработанный прототип как многоязычную (гетерогенную) систему. Последовательность вызова агентов выглядит следующим образом:

- агент извлечения кадров и аудио – инструментальный агент, реализованный на Python и C++ и взаимодействующий через модуль subprocess со специализированным сервером;
- агент преобразования аудиомодальности в текст – основан на модели OpenAI Whisper, в частности на её высокопроизводительном C++-форке Whisper.cpp;
- агент выделения сцен – реализован в виде конвейера, включающего модель YOLO11 для детектирования объектов, библиотеку PySceneDetect для сегментации на сцены и библиотеку pHash для удаления дублирующихся кадров;
- агент преобразования сцен в текст – основан на модели MistralOCR;
- агент агрегации текстов обеих модальностей – реализован на базе модели Mistral Large.

Полученные метрики прототипа

Обучение и валидация всех моделей проводилось с использованием представленного в предыдущих пунктах набора данных, за исключением модели OCR:

- агент, извлекающий кадры и аудио, не требует применения модели машинного обучения, так как является агентом-инструментом, основанным на детерминированном алгоритме;
- для агента «Голос в текст» (модель OpenAI Whisper) в официальном GitHub-репозитории указана ключевая метрика WER порядка 5,8 % для версии Large; в рамках данного исследования для модели Medium величина WER составила 16,3 %, валидация модели производилась путем автоматизированного сравнивания эталонного текста с распознанным;
- для агента обработки видеоряда, реализованного в виде конвейера YOLO11 – PySceneDetect – pHash, значения метрик были получены путем сопоставления автоматически выделенных сцен с эталонной разметкой, сформированной вручную. В результате интегральные метрики качества составили:

Precision = 0,94,

Recall = 1,00,
F1-Score = 0,97;

- для агента оптического распознавания символов (OCR) общая точность распознавания составила 72,8 %, данная метрика основана на показателях из бенчмарков OmniOCR.

Реализованный прототип мультиагентной системы продемонстрировал принципиальную возможность и эффективность подхода, основанного на декомпозиции сложной задачи мультимодального преобразования видеолекций в текст и координации узкоспециализированных агентов. Ключевым результатом является успешная интеграция разнородных компонентов (алгоритмов компьютерного зрения, моделей распознавания речи и текста, языковых моделей) в единый конвейер под управлением агента-оркестратора.

Полученные метрики производительности отдельных модулей свидетельствуют о высоком качестве обработки: конвейер выделения сцен (F1-Score: 0.97) и агент OCR (точность ~95% по бенчмаркам) показали отличные результаты, в то время как модель Whisper (WER: 16.3%) обеспечила приемлемое, но имеющее потенциал для улучшения, качество расшифровки речи.

Важным достижением является разработка гибкой, реконфигурируемой архитектуры, где план обработки генерируется большой языковой моделью на основе конфигурационных файлов, что открывает путь для легкого расширения системы и адаптации к новым типам видеолекций. Основным ограничением текущей версии прототипа является поддержка лишь одного из трех запланированных форматов лекций, что определяет ключевое направление для дальнейшего развития — масштабирование системы и повышение универсальности.

Заключение

На данном этапе исследования:

- проведён анализ теоретических основ мультиагентных систем и паттерна «оркестратор–исполнитель»;
- сформирован алгоритм мультимодальной обработки видеолекции, адаптированный к архитектуре мультиагентной системы;
- выполнены эксперименты по подготовке и оценке моделей машинного обучения, применяемых в агентах системы;
- разработана общая схема взаимодействия агентов и оркестратора для решения поставленной задачи;
- реализован оркестратор, обеспечивающий приём задачи от окружения, её декомпозицию, распределение подзадач между агентами и координацию их работы;
- разработаны агенты, позволяющие обрабатывать видеолекции, в которых преподаватель занимает одну треть кадра, а оставшаяся часть используется для демонстрации слайдов, что обеспечивает полный цикл обработки данного типа видеолекций.

В дальнейшем планируется:

- расширение состава мультиагентной системы за счёт увеличения числа агентов, ориентированных на обработку видеолекций других типов;
- повышение производительности и качества работы уже реализованных агентов.

Благодарности и поддержка

Автор выражает благодарность своему научному руководителю д-ру техн. наук, проф. Андрею Витальевичу Мельникову за всестороннюю поддержку, ценные консультации и неоценимую помощь на всех этапах проведения исследования и подготовки данной статьи. Его знания и опыт, профессиональное руководство и конструктивная критика стали основой для успешной реализации данной работы. Автор также отмечает работы [Бур25, Рез25, Исм25], повлиявшие на данное исследование.

СПИСОК ЛИТЕРАТУРЫ | REFERENCES

[Agr25] Agrawal K., Nargund N. Neural Orchestration for Multi-Agent Systems: A Deep Learning Framework for Optimal Agent Selection in Multi-Domain Task Environments // arXiv. 2025. — arXiv: 2503.04479.

[Ata24] Ataallah K., Shen X., Abdelrahman E., Sleiman E., Zhu D., Ding J., Elhoseiny M. MiniGPT4-Video: Advancing Multimodal LLMs for Video Understanding with Interleaved Visual-Textual Tokens // arXiv. 2024. — arXiv: 2404.03413.

[Dor18] Dorri A., Kanhere S. S., Jurdak R. Multi-Agent Systems: A Survey // IEEE Access. 2018. Vol. 6. P. 28573–28593. EDN: YGWJGH.

[Fou24] Fourney A., Bansal G., et al. Magentic-One: A Generalist Multi-Agent System for Solving Complex Tasks // arXiv. 2024. — arXiv: 2411.04468.

[Gon23] Gonzalez H., Jin H., Baker R., et al. Automatically Generated Summaries of Video Lectures // Proceedings of the 2023 Workshop on Natural Language Generation, Evaluation, and Metrics (BEA). 2023.

[Li24] Li Y., Jiang S., et al. Uni-MoE: Scaling Unified Multimodal LLMs With Mixture of Experts // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2024. Vol. 47. P. 3424–3439.

[Luo20] Luo H., Ji L., et al. UniVL: A Unified Video and Language Pre-Training Model for Multimodal Understanding and Generation // arXiv. 2020. — arXiv: 2002.06353.

[Mil25] Milev I., Balunovi'c M., et al. ToolFuzz - Automated Agent Tool Testing // arXiv. 2025. — arXiv: 2503.04479.

[Tak24] Takeuchi M., Ito A., Nose T. Selection of key sentences from lecture video transcription and its application to feedback to the learner // Proc. 8th Int. Conf. Education and Multimedia Technology ICEMT'2024. Tokyo, Japan, 2024. P. 22–24.

[Wan23] Wang Y., He Y., et al. InternVid: A Large-scale Video-Text Dataset for Multimodal Understanding and Generation // arXiv. 2023. — arXiv: 2307.06942.

[Wu23] Wu J., Gan W., et al. Multimodal Large Language Models: A Survey // Proc. 2023 IEEE Int. Conf. on Big Data (BigData). 2023. P. 2247–2256.

[Xu24] Xu M., Yin W., et al. A Survey of Resource-efficient LLM and Multimodal Foundation Models // arXiv. 2024. — arXiv: 2401.08092.

[Yan23] Yang X., Huang S., et al. Learning Graph-Enhanced Commander-Executor for Multi-Agent Navigation // arXiv. 2023. — arXiv: 2302.04094. P. 1652–1660.

[Ye23] Ye Q., Xu H., et al. mPLUG-Owl: Modularization Empowers Large Language Models with Multimodality // arXiv. 2023. — arXiv: 2304.14178.

[Zou24] Zou H., Luo T., et al. From Seconds to Hours: Reviewing MultiModal Large Language Models on Comprehensive Long Video Understanding // arXiv. 2024. — arXiv: 2409.18938.

[Бур25] Буреев А. С., Антонов В. В., Сапожников А. Ю. Метод валидации электронной конструкторской документации с использованием API КОМПАС-3D // СИИТ. 2025. Т. 7, № 4(23). С. 49-57. EDN: SSNPDM. [[Bureev A. S., Antonov V. V., Sapozhnikov A. Yu. Method of validation of electronic design documentation using KOMPAS-3D API // SIIT. 2025. Vol. 7, No. 4(23). P. 49-57. (In Russian).]]

[Исм25] Исмагулов М. Е. Конвейерный мультимодальный нейросетевой метод обработки видео // СИИТ. 2025. Т. 7, № 1(20). С. 78-85. EDN: TDHVVF. [[Ismagulov M. E. Conveyor-based multimodal neural network method for video processing // SIIT. 2025. Vol. 7, No. 1(20). P. 78-85. (In Russian).]]

[Рез25] Резников Г. А., Синицын Р. Д., Шулик А. М. Современные архитектуры нейронных сетей для тегирования и аннотирования изображений: достижения, вызовы и перспективы // СИИТ. 2025. Т. 7, № 2(21). С. 78-85. EDN: TJFUGV. [[Reznikov G. A., Sinitsyn R. D., Shulik A. M. Modern neural network architectures for image tagging and annotation: achievements, challenges and prospects // SIIT. 2025. Vol. 7, No. 2(21). P. 78-85. (In Russian).]]

ОБ АВТОРАХ | ABOUT THE AUTHORS

ИСМАГУЛОВ Милан Ерикович
Югорский государственный университет, Россия.
m_ismagulov@ugrasu.ru ORCID: 0009-0007-3280-5259.
Аспирант по направлению «Системный анализ управление и обработка информации, статистика».

ISMAGULOV Milan Erikovich
Yugra State University, Russia.
m_ismagulov@ugrasu.ru ORCID: 0009-0007-3280-5259.
PhD student in «System Analysis, Information Management and Processing, Statistics».

МЕТАДАННЫЕ | METADATA

Заглавие: Мультиагентная система для решения задачи мультимодального преобразования видеолекции в текстовый документ.

Авторы: Исмагулов М. Е.

Title: Multi-agent system for solving the problem of multimodal video lecture-to-text transformation.

Authors: Ismagulov M. E.

Аннотация: В статье рассматривается мультиагентная система, разработанная для автоматизированного преобразования видеолекций в полноценный текстовый конспект, который отражает содержание исходного видеофайла с учетом информации из аудио- и видеомодальностей. Целью исследования является создание решения для автоматической трансформации образовательного видеоконтента в структурированный текст, пригодный для дальнейшего использования в учебных целях. В работе применяются методы машинного обучения, включая глубокие нейронные сети, а также подходы мультиагентных систем для координации сложных процессов обработки данных. Разработан прототип системы, реализованный на основе архитектуры «оркестратор-исполнитель». Данная архитектура включает три типа агентов: оркестратор, отвечающий за управление и координацию процессов, агент исполнитель, использующий интеллектуальные модели машинного обучения для анализа контента и агент-инструмент, выполняющий детерминированную обработку данных. На текущем этапе прототип способен обрабатывать один из трех запланированных форматов видеолекций, создавая текстовый документ в формате Markdown. Для обучения и тестирования системы сформирован набор данных, основанный на реальных записях, онлайн-курсов. Метрики прототипа включают: для модели OpenAI Whisper Medium — WER 16,3%; для конвейера YOLO11-PySceneDetect-pHash — Precision: 0.94, Recall: 1.00, F1-Score: 0.97; точность оптического распознавания символов составила 94,89% по результатам бенчмарка. Особенность мультиагентной системы заключается в использовании последовательных конвейеров обработки данных, объединяющих несколько алгоритмов и моделей. В заключение представлены результаты текущего этапа разработки, а также намечены направления дальнейшего совершенствования системы, такие как расширение поддержки форматов и улучшение производительности.

Ключевые слова: Мультимодальная обработка видеолекции; мультиагентные системы; паттерн оркестратор-исполнитель; конвейерная обработка данных; прототип мультиагентной системы.

Язык: Русский.

Статья поступила в редакцию 12 января 2026 г.

Abstract: This article discusses a multi-agent system developed for the automated conversion of video lectures into a comprehensive text summary that reflects the content of the original video file, incorporating information from both audio and visual modalities. The aim of this research is to create a solution for the automatic transformation of educational video content into structured text suitable for further use in academic purposes. The work employs machine learning methods, including deep neural networks, as well as multi-agent system approaches to coordinate complex data processing workflows. A system prototype has been developed based on the "orchestrator-executor" architecture. This architecture includes three types of agents: an orchestrator, responsible for managing and coordinating processes; an executor agent, which utilizes intelligent machine learning models for content analysis; and a tool agent, responsible for deterministic data processing. At the current stage, the prototype can process one of three planned video lecture formats, producing a text document in Markdown format. A dataset based on real online course recordings was compiled for training and testing the system. Prototype metrics include: for the OpenAI Whisper Medium model — a WER of 16.3%; for the YOLO11-PySceneDetect-pHash pipeline — Precision: 0.94, Recall: 1.00, F1-Score: 0.97; Optical Character Recognition accuracy reached 94.89% according to benchmark results. A key feature of the multi-agent system is the use of sequential data processing pipelines, combining multiple algorithms and models. In conclusion, the results of the current development stage are presented, along with directions for further system improvement, such as expanding format support and enhancing performance.

Key words: Multimodal video lecture processing; multi-agent systems; orchestrator-executor pattern; pipeline data processing; multi-agent system prototype.

Language: Russian.

The article was received by the editors on 12 January 2026.