🤖 Бюджетная работа с AI: всего $10 в месяц

🎯 Оптимизация расходов на AI-модели: работаем за $10 в месяц

Ключевые тезисы:

- Стоимость доступа к мощным LLM (Claude, GPT) растёт, стандарт — ~$200/мес.
- Использование AI-агентов для сложных задач значительно увеличивает расход токенов.
- Существует способ получить доступ к качественным моделям за $5-10 в месяц с помощью оптимизации.

💰 Проблема: дорогие тарифы и агенты

Сейчас мы используем не «голые» языковые модели, а **агентов**, которые для решения задач тратят много токенов на фазу размышления и исполнение. Это ведёт к росту стоимости:

- Новая норма — ~$200 в месяц за доступ к топовым моделям.
- К 2026 году цены могут достигать $1000+.
- Многим это не по карману, но потребность в автоматизации и кодинге остаётся.

🚀 Решение: сервис OpenRouter и тариф GO

OpenRouter — платформа-агрегатор моделей.

- **Тариф GO**: $5 в первый месяц, затем $10/мес.
- **Что даёт**: доступ к нескольким качественным моделям (GLM 5.1, Kimi, Minimax, Qwen, DeepSeek).

- Это бюджетная альтернатива, которая подходит для нетяжёлых задач и пет-проектов.

🔧 Уровень 1: Оптимизация входящего контекста (RTK)

RTK AI High Performance CLI Proxy — прокси-сервер между вами и удалённым репозиторием (например, GitHub).

- **Как работает**: фильтрует и удаляет системную информацию из ответов Git (статусы, логи), которая не несёт ценности для модели.
- **Результат**: модель получает чистый, сжатый контекст.
- **Экономия токенов**: от 80% (на командах `ls`, `find`) до 90% (на `cargo test`, `npm test`).
- Устанавливается в OpenRouter одной командой: `rtk init --openrouter`.

🗿 Уровень 2: Оптимизация исходящих ответов (Caveman)

Caveman — *скилл* (навык), заставляющий модель общаться предельно лаконично, «как пещерный человек».

- **Суть**: модель генерирует односложные, сверхкороткие ответы, сохраняя суть.
- **Уровни сжатия**: от минимального до ультра-сжатого.
- **Эффект**: радикально сокращает количество исходящих токенов.
- Устанавливается через менеджер скиллов в OpenRouter, можно выбрать для разных задач (основной чат, коммиты, ревью кода).

⚙️ Уровень 3: Настройка агентов и команд

В OpenRouter можно создать и настроить собственных **агентов** под разные задачи, назначая им конкретные модели.

- **Как**: редактирование конфигурационного файла (`~/.openrouter/config.json`).

- **Стратегия**:
  - Для сложных задач (планирование, анализ) назначать «умную» модель (например, GLM 5.1).
  - Для написания кода — более бюджетную и быструю (например, Minimax).
  - Для ревью — снова «умную» модель.
- После настройки конфига требуется перезагрузка OpenRouter.

🧪 Практический пример: создание веб-приложения

1. **Планирование**: Агент `GLM Planer` (на GLM 5.1) создаёт план для приложения, добавляющего пиво на фото через DALL-E 2.
2. **Кодирование**: Агент-кодер (с активированным Caveman) генерирует код по плану, давая краткие ответы.
3. **Экономия**: Вместо многословных рассуждений модель выдаёт сжатые, но понятные сообщения (например, `User peer up me build`).

✅ Выводы и итог

1. **Комбинация подходов** (бюджетный тариф + RTK + Caveman + кастомные агенты) позволяет **экономить до 50-60% токенов**.
2. За **$10 в месяц** можно получить полноценный рабочий инструмент для вайпкодинга, автоматизации и пет-проектов.
3. Качество взаимодействия будет ниже, чем с GPT-4, но как **«рабочая лошадка»** решение более чем эффективно.
4. **Потратив 30 минут** на настройку, вы получаете доступ к нескольким сильным моделям (GLM 5.1, DeepSeek-V4-Pro и др.) за символическую плату.
