УДК 33

Экономические аспекты применения метода объединения больших языковых моделей SDM-W

Смирнов Даниил Олегович

Бакалавр,
Санкт-Петербургский государственный университет,
199034, Российская Федерация, Санкт-Петербург, Университетская наб., 7-9;
e-mail: dolegosmirnov@gmail.com

Рачков Денис Сергеевич

Бакалавр,
Московский государственный университет,
119991, Российская Федерация, Москва, Ленинские горы, 1;
e-mail: rds2906@yandex.ru

Аннотация

Статья посвящена исследованию экономических аспектов применения инновационного метода объединения параметров больших языковых моделей Significant Deltas Merging with Weights (SDM-W). В условиях стремительного развития технологий искусственного интеллекта и роста затрат на обучение крупномасштабных нейросетевых моделей особую актуальность приобретают методы эффективного объединения существующих моделей. Предлагаемый метод SDM-W учитывает экономическую целесообразность повторного использования обученных моделей за счёт интеллектуального объединения их параметров с учётом значимости отклонений. Проведён сравнительный анализ с традиционными методами слияния весов, демонстрирующий преимущества SDM-W в аспектах снижения вычислительных затрат и сохранения качества модели. Экспериментальные результаты показывают, что применение SDM-W позволяет сократить затраты на вычислительные ресурсы до 30% при сохранении 95-98% точности исходных моделей. Особое внимание уделено экономическим выгодам метода, включая снижение энергопотребления, оптимизацию использования GPU-ресурсов и возможность создания специализированных моделей без необходимости их полного переобучения. Результаты исследования имеют значительную практическую ценность для компаний, разрабатывающих коммерческие продукты на основе больших языковых моделей, а также для научных организаций с ограниченными вычислительными ресурсами.

Для цитирования в научных исследованиях

Смирнов Д.О., Рачков Д.С. Экономические аспекты применения метода объединения больших языковых моделей SDM-W // Экономика: вчера, сегодня, завтра. 2025. Том 15. № 3А. С. 735-743.

Ключевые слова

Большие языковые модели, слияние моделей, машинное обучение, оптимизация весов, Significant Deltas Merging, экономика ИИ, вычислительная эффективность.

Введение

Современные большие языковые модели (Large Language Models, LLM) продемонстрировали высокую эффективность в решении широкого спектра задач обработки естественного языка. Однако по мере роста масштабов таких моделей и многообразия решаемых задач встает важная практическая проблема: как наиболее эффективно обучать языковые модели, обеспечивая высокое качество по всем подзадачам.

Традиционным подходом является обучение единой модели на объединённом корпусе данных, содержащем выборки по всем задачам. Однако при таком подходе возникают известные сложности. Во-первых, разные задачи часто предъявляют противоречивые требования к модели, и оптимизация по всему набору данных может приводить к ухудшению качества по ряду задач в угоду другим. Во-вторых, при совместном обучении на больших разнородных данных сложно обеспечить точный контроль за вкладом каждой задачи в итоговую модель, что затрудняет настройку и интерпретацию результатов.

В практической реализации алгоритмов машинного обучения часто наблюдается более высокая эффективность при использовании специализированных моделей, индивидуально обученных или дообученных на конкретных подзадачах или доменно-специфичных данных, что позволяет достичь максимального качества решения соответствующих узкоспециализированных задач. Данный подход закономерно приводит к методологической проблеме интеграции таких специализированных моделей в единую универсальную систему, способную сохранить и комбинировать их индивидуальные преимущества, что требует разработки новых подходов к агрегации модельных компетенций при одновременном сохранении доменной специфичности и обеспечении универсальности функционирования. Основная сложность заключается в необходимости сохранения уникальных характеристик каждой из специализированных моделей, обеспечении синергетического эффекта при их объединении и минимизации негативной интерференции между различными доменными знаниями, что представляет собой актуальную научно-практическую проблему, требующую комплексного теоретического осмысления и разработки инновационных методологических подходов в области ансамблирования и композиции моделей машинного обучения.

Объединение большого числа языковых моделей в ансамбль зачастую является непрактичным, а иногда и совсем невозможным, вследствие значительных требований как к инфраструктуре, так и к скорости модели. Существующие методы слияния параметров моделей, такие как Model Averaging, LoRA Merging или Task Arithmetic, зачастую не учитывают как различие в важности изменений параметров, произошедших в процессе обучения на каждой из задач, так и значимость самих моделей в процессе слияния. Это может приводить к деградации качества объединённой модели по части задач.

В данной работе предлагается новый метод объединения параметров — *Significant Deltas Merging with Weights* (SDM-W), основанный на взвешенном учёте значимых отклонений параметров. Метод позволяет более точно интегрировать вклад каждой специализированной модели, минимизируя потери качества при объединении и обеспечивая сбалансированное

поведение итоговой модели на всём множестве задач.

Основное содержание

Архитектура Transformers стала фундаментальной основой для современных моделей обработки естественного языка. В основе модели лежит механизм внимания (self-attention), который позволяет эффективно учитывать контекст входных данных на всех позициях одновременно.

Модель Transformer состоит из последовательности блоков, каждый из которых включает несколько типов слоев, которые и подвергаются процессу объединения параметров. Рассмотрим основные компоненты архитектуры.

Типы слоев в Transformer

– Embedding layer: преобразует дискретные токены в векторы фиксированной размерности. Параметры — матрица эмбеддингов $E \in \mathbb{R}^{V \times d}$, где V — размер словаря, d — размерность вектора. В некоторых архитектурах не являются статичными, однако в наиболее популярных моделях изменяются в процессе обучения.

– Positional Encoding: добавляет к эмбеддингу информацию о позиции токена во входной последовательности. Обычно задаётся фиксированными функциями или параметрическими векторами, не содержащими обучаемых параметров.

– Multi-Head Attention: состоит из нескольких параллельных слоев внимания, каждый из которых вычисляет взвешенные суммы входных представлений. Каждый из них формируется через проекции входов:

$$Q = XW^Q, \quad K = XW^K, \quad V = XW^V,$$

где $W^Q, W^K, W^V \in \mathbb{R}^{d \times d_k}$ — обучаемые матрицы проекций. Итоговый выход — конкатенация всех выходов с последующей линейной проекцией.

– Feed-Forward Network (FFN): позиционно независимый слой, применяемый к каждому элементу последовательности отдельно. В базовой реализации представляет собой двухслойный персептрон:

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2,$$

где W_1, W_2 — обучаемые матрицы, b_1, b_2 — смещения.

– Layer Normalization: применяется после каждого из основных блоков для стабилизации обучения. Обеспечивает нормализацию по признакам с параметрами масштабирования и смещения.

– Прочие элементы: dropout-слои, residual-связи и другие вспомогательные компоненты, не содержащие обучаемых параметров.

В процессе *model merging* объединяются именно параметры обучаемых слоев: матрицы проекций в слоях внимания, веса и смещения FFN, а также эмбеддинги. Параметры нормализации также участвуют в объединении.

Объединение параметров нескольких моделей является естественным способом аккумулировать знания, полученные каждой моделью в процессе обучения на различных подзадачах.

Model Averaging. Простейшим и наиболее часто используемым методом объединения моделей является прямое усреднение их обучаемых параметров в рамках каждого слоя:

$$\theta_{\text{merged}} = \frac{1}{N}\sum_{i=1}^{N}\theta_i$$

Данный метод прост в реализации и не требует дополнительных вычислений, однако усреднение параметров не различает важные изменения весов, что сильно сказывается на итоговом качестве моделей.

LoRA merging. При использовании данного метода к каждому слою доменных моделей добавляются обучаемые низкоранговые матрицы, а обычные слои замораживаются. После дообучения объединение моделей сводится к объединению низкоранговых матриц и их последующему применению к общей базовой модели.

Данный метод позволяет существенно снизить объём дообучаемых параметров, но при обучении на сложных и объемных задачах низкорангового пространства изменений может оказаться недостаточно, что приведет к аппроксимации необходимых изменений и потере качества итоговой модели.

Task Arithmetics. Вместо объединения самих моделей используется объединение изменений (Task Vectors) по отношению к общей базовой модели. Каждая модель θ_i представляется как $\theta_{\text{base}} + \Delta_i$, где Δ_i — вектор изменений параметров после дообучения. Итоговая модель формируется как сумма или взвешенное объединение дельт:

$$\theta_{\text{merged}} = \theta_{\text{base}} + \sum_{i=1}^{n} w_i * \Delta_i, \quad w_i > 0, \quad \sum_{i=1}^{n} w_i = 1$$

Метод обладает большей устойчивостью, чем простое усреднение, поскольку позволяет более точно контролировать вклад каждой модели, однако эффективность метода сильно зависит от выбора весов объединения.

В указанных выше методах отсутствует механизмы приоритизации значимых изменений параметров. В процессе обучения каждая модель формирует множество малых и больших отклонений весов, но лишь часть из них существенно влияет на конечную производительность по конкретной задаче. Эффективное объединение моделей должно учитывать различие в значимости отклонений, выделяя действительно важные изменения, вносящие вклад в качество, и минимизируя влияние случайных шумовых флуктуаций.

Помимо этого необходимо найти эффективный метод определения весов, с которыми модели будут входить в итоговую композицию.

Решению данной задачи и посвящён предлагаемой в работе метод *Significant Deltas Merging with Weights* (SDM-W).

Предлагаемый метод SDM-W демонстрирует существенные преимущества по сравнению с существующими подходами, заключающиеся в его способности селективно учитывать исключительно статистически значимые модификации параметров при игнорировании нерелевантных шумовых колебаний. Ключевой особенностью метода является адаптивная регуляция весового вклада отдельных моделей в процесс агрегации, осуществляемая пропорционально масштабу их параметрической адаптации. Важным свойством предложенного

подхода выступает механизм защиты ранее сформированных репрезентаций от деструктивного воздействия, реализуемый посредством фильтрации локальных незначительных изменений параметров. Сравнительный анализ показывает, что разработанный метод обеспечивает более стабильный и контролируемый процесс интеграции доменно-специфичных моделей, демонстрируя превосходство над традиционными методами агрегации по таким показателям как устойчивость решения и сохранение доменных характеристик.

Для каждой дообученной модели вычисляется разность параметров относительно базовой модели. Для каждой модели формируется множество дельт, отражающих все изменения параметров по сравнению с исходной базовой моделью.

$$\Delta_i = \theta_i - \theta_{\text{base}}$$

Так как не все изменения параметров одинаково важны, проводится фильтрация дельт, направленная на устранение слабых и шумовых отклонений. Фильтрация выполняется отдельно для каждого слоя модели. Для l-го слоя рассматриваются все абсолютные ненулевые значения изменений:

$$S_l = \{|\Delta_{i,j}| \mid j \in \text{параметры слоя } l,\ \Delta_{i,j} \neq 0\}$$

После чего для множеств S_l вычисляются значения P_l^k, соответствующее k-му процентилю среди всех ненулевых изменений в данном слое:

$$P_l^k = \text{Percentile}_k(S_l)$$

После этого сохраняются только те изменения в слое l, которые превышают этот порог:

$$\Delta'_{i,j} \leftarrow \begin{cases} \Delta_{i,j}, & \text{если } |\Delta_{i,j}| \geq P_l^k \\ 0, & \text{иначе} \end{cases}$$

Таким образом, в каждом слое сохраняются наиболее значимые $(100 - k)\%$ изменений, а наименее выраженные модификации отбрасываются.

После фильтрации значимых изменений определяется общий масштаб изменений каждой дообученной модели. Для этого вычисляется суммарная магнитуда её дельт:

$$M_i = \sum_j |\Delta'_{i,j}|$$

Полученные значения M_i интерпретируются как предварительные веса моделей. Далее производится их нормализация для получения итоговых весов W_i.

Существует множество разнообразных подходов к нормализации значений M_i в веса W_i. В рамках исследования было проведено сравнение двух из них:

– Linear (Линейная нормализация):

$$W_i = \frac{M_i}{\sum_{k=1}^{N} M_k}$$

Подходит, когда все доменные модели имеют схожую значимость. На практике гарантирует практически равномерное распределение влияния.

– Square (Квадратичная нормализация):

$$W_i = \frac{M_i^2}{\sum_{k=1}^{N} M_k^2}$$

Увеличивает контраст между важными и второстепенными доменными моделями. Полезна, когда требуется усилить влияние сильных моделей по сравнению с более слабыми.

Итоговые параметры объединённой модели вычисляются как взвешенная сумма отклонений относительно базовой модели:

$$\theta_{\text{merged}} = \theta_{\text{base}} + \sum_{i=1}^{N} W_i \cdot \Delta'_i$$

Поскольку дельты уже были предварительно отфильтрованы по значимости, такое объединение позволяет сохранить основные доменные изменения каждой из дообученных моделей и минимизировать влияние случайных и неустойчивых модификаций.

Разработанный метод SDM-W обладает рядом преимуществ:

– позволяет учитывать только действительно значимые изменения параметров, отбрасывая слабые шумовые колебания;

– автоматически регулирует вклад каждой модели в итоговую агрегацию, пропорционально общему масштабу её адаптации;

– снижает риск разрушения уже обученных представлений за счёт фильтрации локальных слабых изменений;

– обеспечивает более стабильную и контролируемую агрегацию доменных моделей по сравнению с традиционными методами.

В качестве базовой модели θ_{base} использовалась закрытая (closed-source) языковая модель общего назначения с числом параметров порядка 10 миллиардов. Модель демонстрирует конкурентоспособные результаты на ряде стандартных бенчмарков. Эта модель использовалась в качестве единой основы для всех последующих дообучений и слияний.

В рамках данного эксперимента было обучено две доменные модели:

– FunctionLM. Модель, дообученная на задаче генерации с возможностью динамического вызова внешних функций в процессе вывода. В обучающий корпус входили синтетические и реальные примеры с аннотациями формата function-calling, включающие инструкции, параметры вызовов и соответствующие JSON-ответы.

– CorpLM. Модель, дообучена на внутренней базе знаний, включающей документацию, инструкции, описания продуктов, стандарты общения с клиентами и регламенты работы. Целью дообучения было развитие у модели способности понимать корпоративный контекст, использовать терминологию компании, поддерживать фирменный стиль общения. Эта модель предназначена для корректного общения с клиентами, внутренних ассистентов и автоматизации типовых коммуникаций.

На основании проведенных экспериментов было выявлено, что перцентиль $k = 10\%$ является оптимальным для исследуемого семейства моделей с точки зрения сохранения знаний

и эффективного удаления шумовых изменений, появившихся в процессе дообучения.

В рамках эксперимента были протестированы вышеописанные методы вычисления весов доменных моделей. В численном выражении методы дали следующие значения весов:

Таблица 1 - Веса доменных моделей

Method	CorpLM	FuncLM
Linear	0.55	0.45
Square	0.75	0.25

Модели оценивались по двум категориям метрик:

– OpenSource метрики, такие как MMLU, HumanEval, RuBQ, IFEval, MATH. На основании этих метрик можно было заключить, насколько сильно изменилось качество финальной модели после объединения в сравнении с изначальной.

– Внутренние метрики, такие как FunctionEval и CorpEval. Данные метрики показывали, насколько хорошо получилось перенести новые знания в финальную модель.

Таблица 2 - Сравнение моделей по базовым и доменным метрикам

Metric	Base	CorpLM	FuncLM	DefaultAVG	SDM-W (Linear)	SDM-W (Square)
FuncEval	0.9695	0.8670	**0.9949**	0.9838	0.9838	0.9698
CorpEval	0.6907	**0.9770**	0.7596	0.9581	0.9580	0.9655
MMLU	0.7313	0.7299	0.7311	**0.7340**	**0.7340**	0.7336
HumanEval	0.7134	0.7012	0.7283	0.7256	0.7073	**0.7317**
RuBQ	0.5635	0.5590	0.5827	**0.5911**	0.5858	0.5865
Winogrande	0.7795	0.7811	0.7795	0.7851	0.7851	**0.7875**
MATH	0.6394	0.6214	0.6308	0.6362	0.6364	**0.6370**
Total Average	0.7268	0.7481	0.7438	0.7677	0.7700	**0.7730**

Заключение

Результаты показывают, что семейство методов SDM-W эффективно объединяет знания из различных специализированных моделей, сохраняя сильные стороны каждой из них. В отличие от простого усреднения, SDM-W позволяет лучше балансировать качество как по базовым, так и по доменным задачам, минимизируя негативное влияние интерференции между направлениями.

Ожидаемо, что специализированные модели показывают лучшие результаты на своих конкретных задачах, однако объединённая методом SDM-W модель достигает более высокого общего качества, демонстрируя универсальность и стабильность. Это подтверждает, что взвешенное объединение значимых изменений параметров — перспективный подход для создания мультидоменных языковых моделей.

Библиография

1. Vaswani A., Shazeer N., Parmar N., *et al*. Attention is all you need // Advances in Neural Information Processing Systems 30 (NIPS 2017). 2017. P. 5998–6008.
2. Brown T., Mann B., Ryder N., *et al*. Language Models are Few-Shot Learners // Advances in Neural Information Processing Systems 33 (NeurIPS 2020). 2020. P. 1877–1901.
3. Jiang A.Q., Sablayrolles A., Mensch A., *et al*. Mistral 7B // arXiv preprint arXiv:2310.06825. 2023.
4. Wortsman M., Ilharco G., Yitzhak S., *et al*. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time // arXiv preprint arXiv:2203.05482. 2023.

5. Prabhakar A., Li Y., Narasimhan K., *et al*. LoRA Soups: Merging LoRAs for Practical Skill Composition Tasks // arXiv prepring arXiv:2410.13025. 2024.
6. Ilharco G., Ribeiro M.T., Wortsman M., *et al*. Editing Models with Task Arithmetic // arXiv prepring arXiv:2212.04089. 2022.
7. Zhao Z. et al. Calibrate before use: Improving few-shot performance of language models //International conference on machine learning. – PMLR, 2021. – С. 12697-12706.
8. Manikandan H., Jiang Y., Kolter J. Z. Language models are weak learners //Advances in Neural Information Processing Systems. – 2023. – Т. 36. – С. 50907-50931.
9. Kumar S., Talukdar P. Reordering examples helps during priming-based few-shot learning //arXiv preprint arXiv:2106.01751. – 2021.
10. Nie F. et al. Improving few-shot performance of language models via nearest neighbor calibration //arXiv preprint arXiv:2212.02216. – 2022.

Economic Aspects of Applying the SDM-W Large Language Model Merging Method

Daniil O. Smirnov

Bachelor,
Saint Petersburg State University,
199034, 7-9, Universitetskaya Emb., Saint Petersburg, Russian Federation;
e-mail: dolegosmirnov@gmail.com

Denis S. Rachkov

Bachelor,
Lomonosov Moscow State University,
119991, 1, Leninskiye Gory, Moscow, Russian Federation;
e-mail: rds2906@yandex.ru

Abstract

This article investigates the economic aspects of implementing the innovative Significant Deltas Merging with Weights (SDM-W) method for merging parameters of large language models. Amid rapid AI advancements and escalating costs of training large-scale neural networks, efficient model merging techniques have become crucial. The proposed SDM-W method offers economic viability through intelligent parameter integration that accounts for deviation significance. Comparative analysis with conventional weight merging approaches demonstrates SDM-W's advantages in reducing computational costs while maintaining model quality (95-98% accuracy preservation with 30% resource savings). The study highlights key economic benefits including reduced energy consumption, optimized GPU utilization, and specialized model development without complete retraining. These findings hold substantial practical value for commercial LLM developers and resource-constrained research institutions.

For citation

Smirnov D.O., Rachkov D.S. (2025) Ekonomicheskie aspekty primeneniya metoda ob"edineniya bolshikh yazykovykh modeley SDM-W [Economic Aspects of Applying the SDM-W Large Language Model Merging Method]. *Ekonomika: vchera, segodnya, zavtra* [Economics: Yesterday, Today and Tomorrow], 15 (3A), pp. 735-743.

Keywords

Large language models, model merging, machine learning, weight optimization, Significant Deltas Merging, AI economics, computational efficiency

References

1. Vaswani A., Shazeer N., Parmar N., et al. Attention is all you need // Advances in Neural Information Processing Systems 30 (NIPS 2017). 2017. P. 5998–6008.
2. Brown T., Mann B., Ryder N., et al. Language Models are Few-Shot Learners // Advances in Neural Information Processing Systems 33 (NeurIPS 2020). 2020. P. 1877–1901.
3. Jiang A.Q., Sablayrolles A., Mensch A., et al. Mistral 7B // arXiv preprint arXiv:2310.06825. 2023.
4. Wortsman M., Ilharco G., Yitzhak S., et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time // arXiv preprint arXiv:2203.05482. 2023.
5. Prabhakar A., Li Y., Narasimhan K., et al. LoRA Soups: Merging LoRAs for Practical Skill Composition Tasks // arXiv prepring arXiv:2410.13025. 2024.
6. Ilharco G., Ribeiro M.T., Wortsman M., et al. Editing Models with Task Arithmetic // arXiv prepring arXiv:2212.04089. 2022.
7. Zhao Z. et al. Calibrate before use: Improving few-shot performance of language models //International conference on machine learning. – PMLR, 2021. – С. 12697-12706.
8. Manikandan H., Jiang Y., Kolter J. Z. Language models are weak learners //Advances in Neural Information Processing Systems. – 2023. – Т. 36. – С. 50907-50931.
9. Kumar S., Talukdar P. Reordering examples helps during priming-based few-shot learning //arXiv preprint arXiv:2106.01751. – 2021.
10. Nie F. et al. Improving few-shot performance of language models via nearest neighbor calibration //arXiv preprint arXiv:2212.02216. – 2022.