Возможности участия LLM в процессе настройки параметров солверов

Устюгов В.Н., Институт математики им. С.Л. Соболева СО РАН (Омский филиал)

Исследование выполнено за счет гранта Российского научного фонда № 22-71-10015, https://rscf.ru/en/project/22-71-10015/

Цель

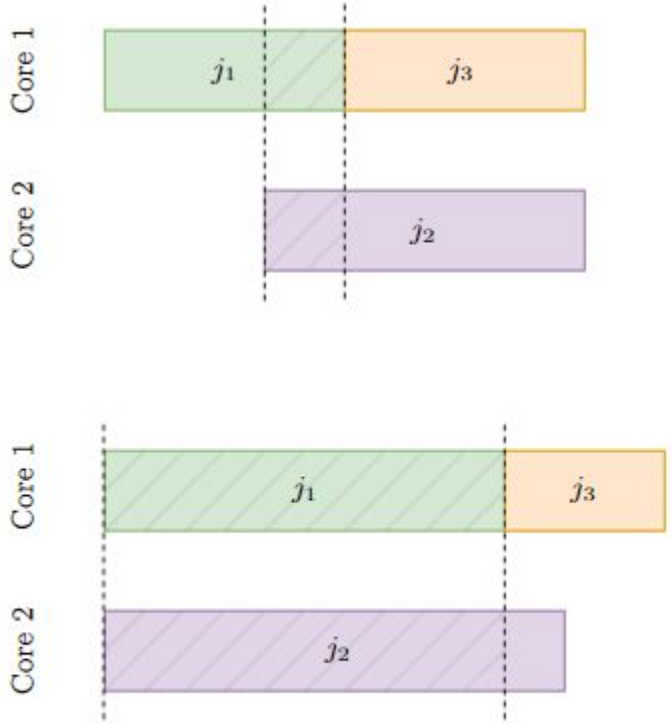
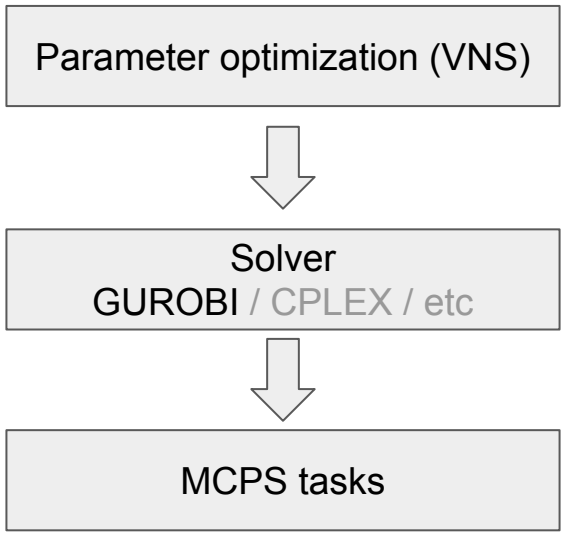
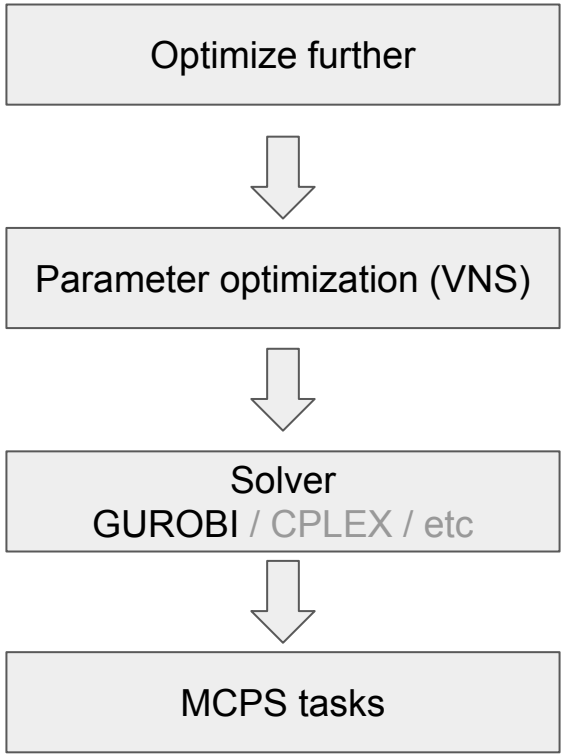


Рис. 1. Выполнение работ с задержкой (сверху) и без задержки (снизу)

Процесс


Parameter optimization (VNS)
Solver
GUROBI / CPLEX / etc
MCPS tasks

Процесс


Optimize further
Parameter optimization (VNS)
Solver
GUROBI / CPLEX / etc
MCPS tasks

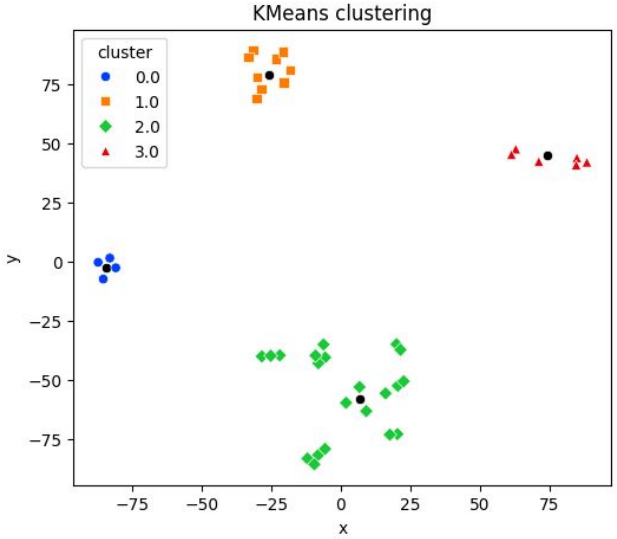
Зачем оптимизировать?

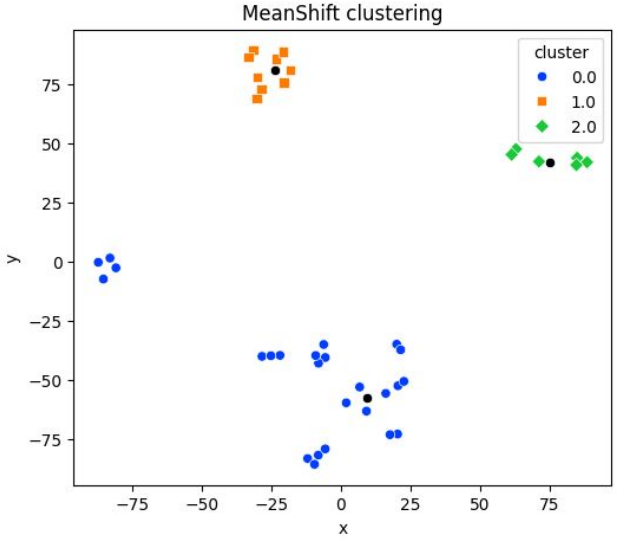
Таблица 1. Среднее время работы пакета CPLEX

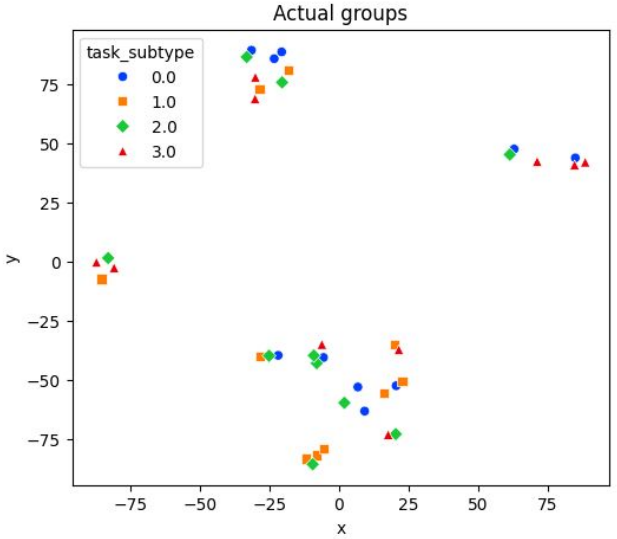
Table 1. Average time of CPLEX package

	4 jobs	6 jobs	**7 jobs**	**8 jobs**	**10 jobs**
Trivial order	0.4 s	4.3 min	13 min	16 min	15.5 min
One to many to one	0.2 s	3 s	26 s	6.3 min	14.8 min
Random order	0.2 s	3.6 s	18 s	32 s	3.6 min
Bitree order	0.2 s	4 s	1.5 min	7.2 min	16 min

Как оптимизировать?



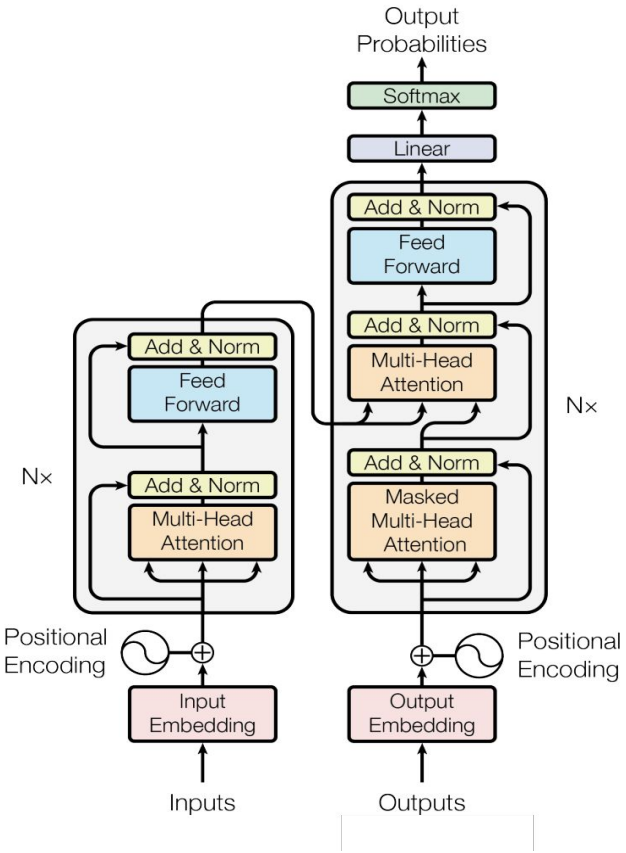




LLM (Large Language Models)

BERT

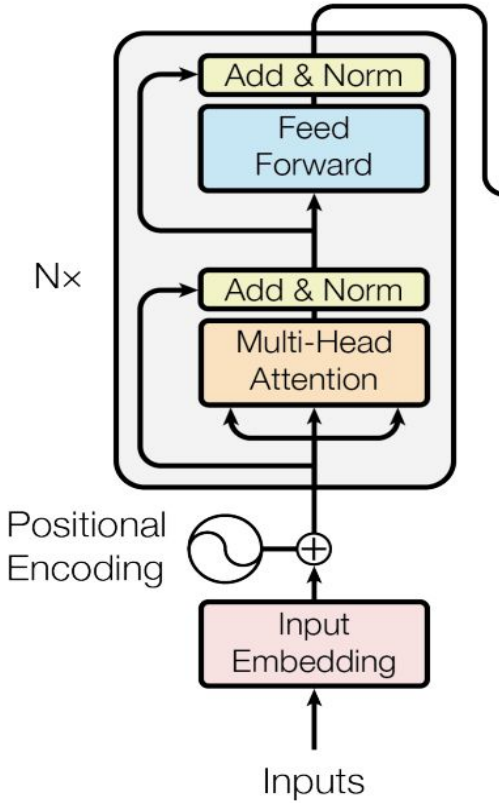
Encoder



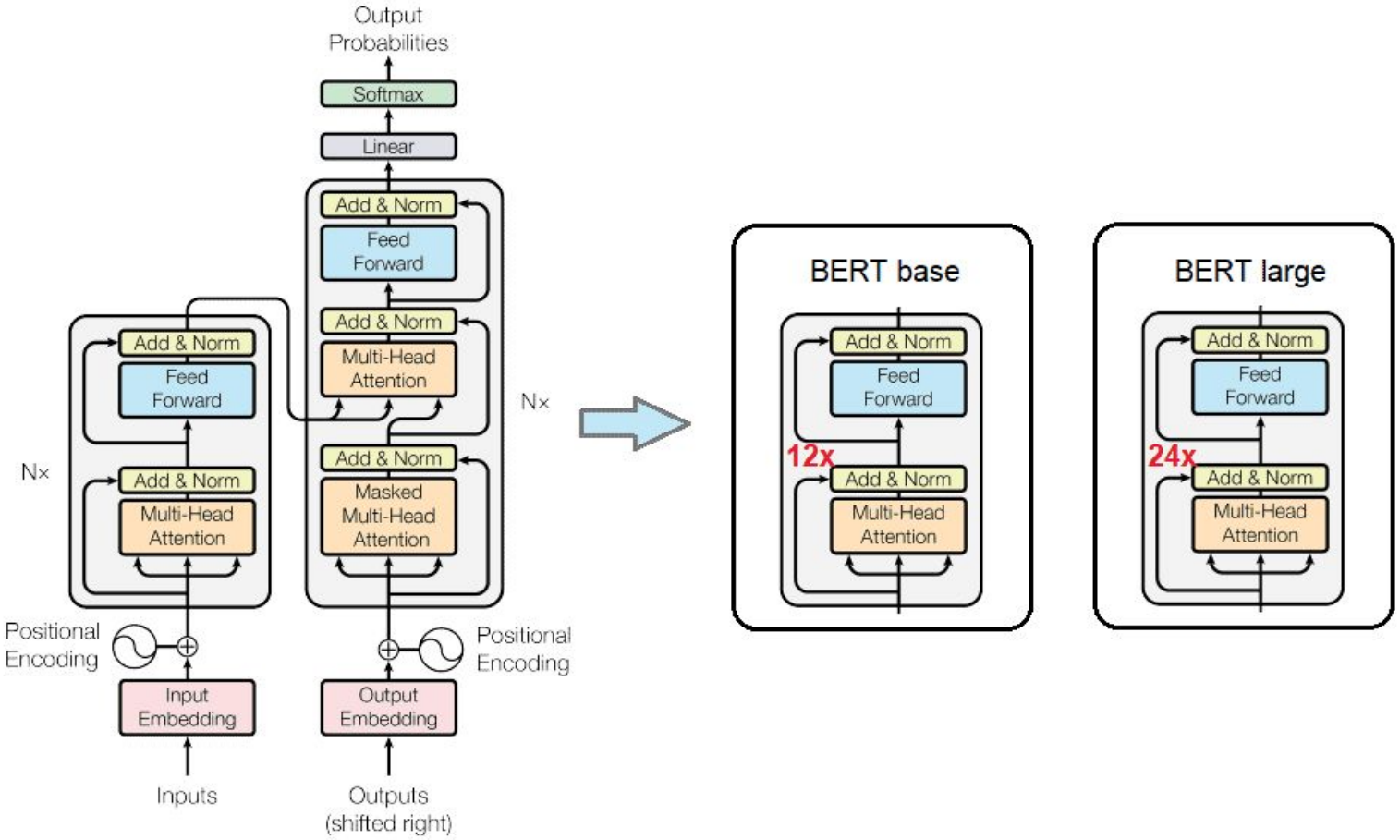
GPT

Decoder

LLM (Large Language Models)



BERT(Bidirectional Encoder Representations from Transformers)



BERT(Bidirectional Encoder Representations from Transformers)

Предобучение

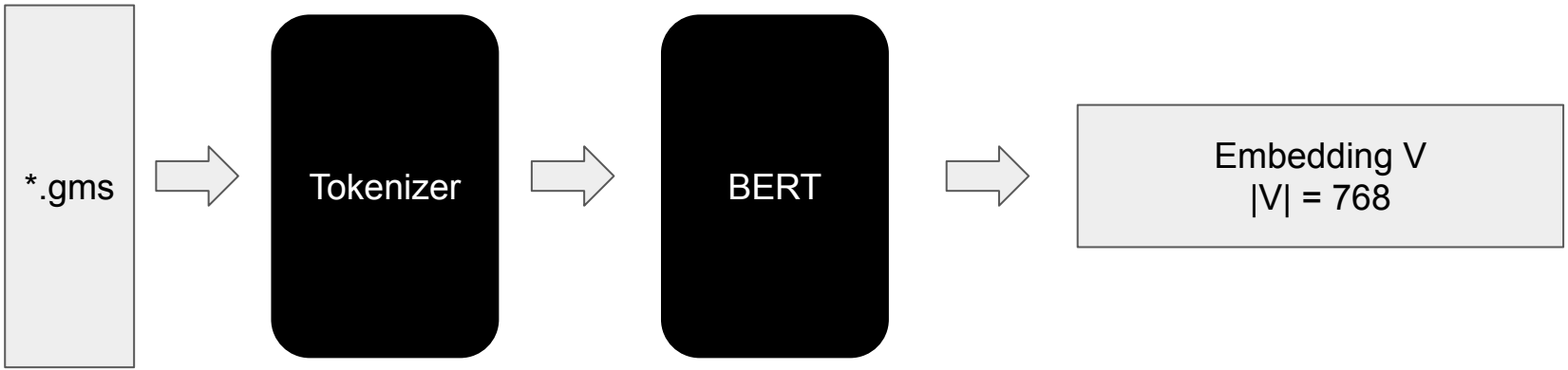
BERT обучается одновременно на двух задачах — предсказания следующего предложения (англ. next sentence prediction) и генерации пропущенного токена (англ. masked language modeling). На вход BERT подаются токенизированные пары предложений, в которых некоторые токены скрыты. Таким образом, благодаря маскированию токенов, сеть обучается глубокому двунаправленному представлению языка, учится понимать контекст предложения. Задача же предсказания следующего предложения есть задача бинарной классификации — является ли второе предложение продолжением первого. Благодаря ей сеть можно обучить различать наличие связи между предложениями в тексте.Интерпретация этапа предобучения — обучение модели языку.

BERT(Bidirectional Encoder Representations from Transformers)

Точная настройка (Fine-tuning)

Этот этап обучения зависит от задачи, и выход сети, полученной на этапе предобучения, может использоваться как вход для решаемой задачи. Так, например, если решаем задачу построения вопросно-ответной системы, можем использовать в качестве ответа последовательность токенов, следующую за разделителем предложений. В общем случае дообучаем модель на данных, специфичных задаче: знание языка уже получено на этапе предобучения, необходима лишь коррекция сети.Интерпретация этапа fine-tuning — обучение решению конкретной задачи при уже имеющейся общей модели языка.

BERT(Bidirectional Encoder Representations from Transformers)



Что делать с векторными представлениями?

$P = (p_1,...,p_k)$ - вектор параметров солвера (далее *конфигурация*)

Задача регрессии:

$\mathbf{P_i^* = V_i W}$, $i = 1,...,J$, где J это число различных индивидуальных задач

$V_i = \text{BERT}(\text{Tokenizer}(I_i))$

Как оптимизировать?

Full task set
5 different dimensionalities, ~200 samples each

1 dim/all samples

random dims/ k% of samples

1 dim/ k% of samples

The rest

Спасибо за внимание

Литература:

1. А.В.Еремеев, М.Ю.Сахно, *Построение расписания для многоядерного процессора с учетом взаимного влияния работ*

2. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*

Запасная страница 1