



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М. В. ЛОМОНОСОВА  
ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ  
КАФЕДРА МАТЕМАТИЧЕСКИХ МЕТОДОВ ПРОГНОЗИРОВАНИЯ

Алексеев Илья Алексеевич

**Контрастивное обучение с аугментациями для построения векторного  
представления диалога**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**Научный руководитель:**  
научный сотрудник  
Кравцова Ольга Анатольевна

**Научный консультант:**  
Кузнецов Денис Павлович

# Оглавление

1	Введение . . . . .	4
2	Постановка задачи . . . . .	6
	2.1    Диалоговые данные . . . . .	6
	2.2    Аугментация . . . . .	6
	2.3    Векторное представление . . . . .	7
3	Обзор литературы . . . . .	8
	3.1    Векторизация текста . . . . .	8
	3.2    Контрастивное обучение . . . . .	8
	3.3    Похожие методы . . . . .	9
4	Метод . . . . .	11
	4.1    Аугментации . . . . .	11
	4.2    Наборы аугментаций . . . . .	12
	4.3    Дообучение кодировщика . . . . .	12
	4.4    Оценивание . . . . .	13
5	Эксперименты . . . . .	15
	5.1    Набор данных . . . . .	15
	5.2    Оптимизация набора аугментаций . . . . .	15
	5.3    Анализ разброса . . . . .	17
	5.4    Анализ инвариантности . . . . .	18
	5.5    Анализ скрытых состояний . . . . .	19
6	Заключение . . . . .	20
	<b>Список литературы</b>	<b>21</b>
1	Вспомогательные модели . . . . .	27
	1.1    Близость между репликами в диалоге . . . . .	27
	1.2    Списочная модель . . . . .	29
2	Композиции аугментаций . . . . .	30

## **Аннотация**

Векторные представления, полученные из предварительно обученных языковых моделей, доказали свою чрезвычайную эффективность для различных текстовых задач, таких как классификация пар текстов, оценка близости и поиск. Соответствующие модели обычно обучаются на больших объемах чистых и разнообразных данных с использованием контрастивной функции потерь. К сожалению, для области диалоговых систем нет таких наборов данных. В данной работе описан процесс создания синтетического набора диалогов с помощью специально разработанных аугментаций. Проведено сравнение разных аугментаций с точки зрения качества переноса знаний на задачи, связанные с диалоговыми системами.

# 1. Введение

Получение векторных представлений является одной из ключевых задач в машинном обучении и становится все более популярной в последние годы, поскольку вектор является удобным математическим объектом и латентным признаковым описанием. Если векторизация точно и всесторонне кодирует семантику исходных данных, это открывает возможность её использования в широком спектре задач.

В области обработки естественного языка давно известны классические методы векторизации текста, такие как мешок слов [1] и tf-idf [2]. Благодаря глубокому обучению, были получены методы векторизации слов, такие как word2vec [3] и GloVe [4], передающих семантическую близость между словами, а также CoVe [5] и ELMo [6], передающие информацию о слове с учетом контекста, в котором оно встретилось. Недавно появились мощные модели-кодировщики, которые создают универсальные векторные представления текста [7—9]. Они включают в себя настолько много всесторонней семантической информации о текстах, что их можно без дообучения применять к самым разным задачам, таким как классификация, кластеризация, ранжирование, семантическая текстовая близость и многим другим [10]. Успех этих моделей в значительной степени обусловлен использованием контрастивного обучения на огромных наборах данных [11].

Однако чем более специфична структура данных, тем сложнее собрать качественную и разнообразную обучающую выборку достаточно большого размера. Именно так обстоит дело в домене диалоговых данных, поэтому большинство диалоговых моделей строят на основе внутритокенного и ме-жтокенного взаимодействия в тексте, но не на основе задач на уровне всего диалога. Так, языковые модели, такие как те, что представлены в [12, 13], утилизируют иерархическую и временную природу диалогов.

С другой стороны, существуют способы синтезировать данные для обучения. На сегодняшний день было разработано множество методов для генерации диалогов [14—18], что полезно для задачи тонкой настройки больших языковых моделей. Однако для обучения языковых моделей-кодировщиков необходимы выборки с парными диалогами и разметкой «похожие/не похожие». Самый простой способ синтезировать пары диалогов — это применить текстовые аугментации [19]. В данной работе мы описываем метод построе-

ния набора данных диалогов для контрастивного обучения с использованием различных техник аугментации.

Вклад этой работы:

- новые методы аугментаций для диалога;
- анализ процедуры контрастивного обучения на аугментациях диалога;
- анализ адаптированности существующих моделей к домену задачеориентированных диалогов.

## 2. Постановка задачи

### 2.1. Диалоговые данные

Диалог определяется следующим списком:

$$d = [(u_1, s_1), \dots, (u_n, s_n)],$$

где  $u_i$  представляет собой высказывание участника  $s_i$  диалога на шаге  $i$ . Существуют так называемые задачеориентированные диалоги с двумя участниками: системой и пользователем. Приблизительно можно описать их как диалоги между работником сферы услуг и клиентом. Во время диалога у клиента есть различные намерения, которые сотрудник стремится распознать и удовлетворить. Например, намерением могут быть поиск ресторана, бронирование столика, вызов такси или покупка билета на поезд. Мы будем считать два диалога схожими, если они имеют похожий набор намерений и принадлежат к одной и той же области.

### 2.2. Аугментация

Аугментацией назовём создание новых валидных объектов путем трансформации существующих. Валидными объектами назовем те, которые достаточно похожи на реальные данные. Пусть  $D$  — множество валидных объектов. Тогда аугментация представляет собой неидентичное отображение  $\text{aug}(d)$ , которое не выводит объекты за пределы множества валидных объектов и сохраняет ключевые характеристики исходного объекта, например, намерения:

$$\text{aug} : D \rightarrow D.$$

Обычно аугментацию реализуют путем внесения небольших изменений в валидные объекты из обучающей выборки. Например, аугментацией изображения является небольшой поворот или размытие. Аугментация текста особенно сложна, потому что валидность предполагает соблюдение языковых правил, осмысленность и принадлежность определенному стилю. В случае задачеориентированных диалогов необходимо сохранять структуру, разграничение ролей и предмет разговора.

## 2.3. Векторное представление

Векторное представление — это отображение  $D$  в векторное пространство:

$$e : D \rightarrow \mathbb{R}^n.$$

Для объекта  $d$  его векторное представление  $e(d)$  должно передавать некоторую семантику  $d$ . Это отражается в том, что  $e(d)$  может содержать лексическую информацию или латентные признаки, полезные для классификации и других задач. Особенно ценно, если использование векторных представлений  $e(a), e(b)$  позволяет оценить семантическую близость объектов  $a, b$  как метрическую близость между их векторизациями.

Мы формулируем нашу задачу как задачу обучения функции  $e_\theta$  для получения векторного представления слова. Такую функцию называют кодировщиком.

## 3. Обзор литературы

### 3.1. Векторизация текста

Один из главных прорывов в построении векторного представления текста возник в результате решения задачи семантической близости текстов [20]. Методы, предшествующие этому, были либо слабыми, такими как усреднение векторных представлений слов без учета контекста, либо требовали неподъемной вычислительной нагрузки, как в случае кросс-кодировщика BERT [21]. Вместо этого было предложено использовать би-кодировщики, которые сравнивают тексты, усредняя скрытые состояния последнего блока трансформера.

### 3.2. Контрастивное обучение

Чтобы заставить кодировщик выдавать богатые векторные представления, необходимо обучать его на задачах, где задействованы семантические признаки. Одной из популярных таких задач является контрастивное обучение:

$$\mathcal{L} = -\log \frac{\exp(\text{sim}(x, y))}{\sum_z \exp(\text{sim}(x, z))}.$$

Здесь  $x = e_\theta(a)$ ,  $y = e_\theta(b)$  — векторные представления пары семантически похожих объектов, а  $z = e_\theta(c)$  — векторизация объекта, семантически удаленного от  $x$ ,  $\text{sim}$  — функция метрической близости (например, косинус). Эта задача настраивает параметры модели  $\theta$  так, чтобы векторные представления передавали семантическую близость в виде косинусной близости.

Данный функционал может быть рассмотрен как частный случай так называемого самообучения, которое впервые обрело популярность в области компьютерного зрения [11, 22—24]. Эти подходы к обучению предложили новый взгляд на обучение без учителя и на задачу переноса знаний (transfer learning), что позволило перейти от обучения моделей к обучению представлений (representation learning), что оказалось более эффективно на практике для различных «вытекающих задач» (downstream tasks).

Один из наиболее известных методов для формирования отрицательных примеров  $z$  (negative sampling) был представлен в работе [3]. Он заключается в случайном разыгрывании объектов из обучающего набора данных.

Современные методы с использованием случайных отрицательных примеров [25] реализуют эту идею, выбирая случайные объекты не из всего набора данных, а из текущего батча во время обучения (*in-batch negative sampling*). Для этого нужны большие вычислительные ресурсы, метод выигрывает от большого батча. Другая идея заключается в использовании тяжелых отрицательных примеров (*hard negative sampling*) [26, 27], которые выбираются как ближайшие к положительной паре среди всех отрицательных примеров.

Наиболее эффективно обучать векторные представления, используя положительные пары, размеченные человеком. Например, наборы данных для задач *natural language inference* [28], ответов на вопросы [29], проверки фактов [30] и перефразирования содержат пары семантически схожих текстов, что делает их подходящими для контрастивного обучения [9]. Эти данные чрезвычайно чисты, но ограничены по объему. Другой подход включает сбор данных с веб-страниц, таких как вопросно-ответные форумы и социальные медиа-платформы. Эти данные представлены в изобилии, но могут содержать шум. Когда нет готовых ресурсов, таких как размеченные или собранные пары, аугментация остается единственным вариантом. В этом случае два аугментированных представления одного и того же объекта должны сохранять основную семантику объекта.

Современные методы построения векторных представлений текста, такие как BGE [7], GTE [9], E5 [8], следуют одному и тому же алгоритму обучения. Во-первых, обучается языковая модель ориентированная на поиск [31, 32]. Во-вторых, проводится процесс основательного предобучения с использованием большого корпуса текстовых пар с контрастным обучением на больших батчах. Наконец, проводится тонкая настройка под множество задач из размеченных наборов текстовых пар с использованием контрастивного обучения с тяжелыми отрицательными примерами.

### 3.3. Похожие методы

**Векторное представление текста.** Существует несколько методов векторизации текста, которые обучены с помощью аугментаций. CERT [33] использует обратный перевод для добычи положительных пар, в то время как ConSERT [34] применяет различные аугментации на уровне токенов. Doc2vecC [35] полагается на аугментации на основе тезауруса WordNet [36]

и обратный перевод.

**Векторное представление диалогов.** Dial2vec [37] и DialogueCSE [38] модифицируют архитектуру трансформера, добавляя кросс-внимание между различными группами высказываний после всех слоев трансформера. Для получения положительной пары оба подхода используют идею самонаставления (self-guidance), аналогичную контрастивному обучению с аугментациями, потому что они получают два представления одного и того же объекта. В Dial2vec это локальная и глобальная информация о диалоге, а в DialogueCSE это высказывания первого собеседника по сравнению с высказываниями второго собеседника.

Наш подход отличается от упомянутых выше методов по следующим причинам:

- Наш набор аугментаций более широк. Он выходит за рамки простого перефразирования и замены синонимов.
- В предыдущих работах нет масштабного обзывационного исследования аугментаций.
- Наши аугментации соответствуют набору допустимых объектов, сохраняя грамматические и значимые конструкции. Они не нарушают структуру диалога, так как они разработаны с использованием контекстно-зависимых языковых моделей, без простого перемешивания и удаления токенов.
- Наш подход не модифицирует архитектуру BERT, т.е. наши результаты могут быть использованы в качестве инициализации в других исследованиях BERT-подобных архитектур.

## 4. Метод

Предлагаемая методология решения задачи построения векторного представления делится на два этапа: 1) разработка аугментаций для диалоговых данных, 2) контрастивное обучение на аугментациях. По итогу первого этапа получается набор данных, который состоит из объектов со следующей схемой: исходный диалог, список различных аугментированных версий.

### 4.1. Аугментации

**Вставка токенов.** Одним из простых, но эффективных способов аугментации текста является его удлинение путем вставки дополнительных токенов. Для этой цели мы добавили специальный токен [MASK] в случайные места в диалогах и использовали модели-трансформеры, обученную на задачу заполнения масок [39]. Вставка отклоняется, если токен, предложенный моделью, является только фрагментом слова [40, 41], или если вероятность предсказания ниже вручную подобранного порога. Чтобы учесть контекст диалога во время вставки токенов, несколько последовательных фраз диалога подаются на вход модели, заполняющей маски, как компромисс между подачей отдельных фраз и подачей всего диалога.

**Замена токенов.** Этот метод идентичен предыдущему, за исключением того, что вместо добавления токена маски мы заменяем некоторые токены в исходном диалоге.

**Обратный перевод.** Перевод из оригинального языка на другой язык и затем обратно на оригинальный язык. Для этой цели использовались модели нейронного машинного перевода [42].

**Перемешивание высказываний.** Предыдущие методы аугментации модифицируют диалог в пределах отдельного высказывания, они применимы к произвольным текстовым данным. Изменение порядка фраз диалога является более сильной аугментацией. Для реализации этой идеи мы предлагаем использовать модель, которая измеряет сходство между фразами в диалоге. Используя эти сходства, можно сгруппировать высказывания внутри каждого диалога с помощью агломеративной кластеризации. Эксперименты показали, что эти группы представляют собой отдельные независимые этапы диалога, которые можно перемешивать между собой.

**Обрезка диалога.** Отдельные группы фраз диалога могут быть отброшены, что приводит к укороченному диалогу с меньшим количеством высказываний.

## 4.2. Наборы аугментаций

Для расширения набора аугментаций еще больше мы использовали простые преобразования «на лету» и композицию нескольких аугментаций. В теории это добавляет вычислительной сложности, однако не обязательно ведет к улучшению итогового результата. Например, композиция аугментаций показала более низкую производительность в предварительных экспериментах. Это приводит к необходимости подбора такого набора аугментаций, который бы достигнул компромисса между вычислительной сложностью и качеством. Наборы аугментаций, которые мы рассматривали, представлены в таблице 1.

имя	вставка	замена	перевод	обрезка	перемеш.	«на лету»
t-l	+	+	-	-	-	-
t-h	+	+	+	-	-	-
a-h	+	+	+	+	+	-
a-l-m	+	+	-	+	+	+
a-l-m-dse	+	+	-	+ <sup>†</sup>	+ <sup>†</sup>	+

Таблица 1: Рассматриваемые наборы аугментаций. Эти названия являются акронимами: t означает тривиальный, l — легкий, h — тяжелый, a — расширенный, m — смешанный с трансформациями «на лету». Тривиальный включает только аугментации на уровне токенов, расширенный — аугментации как на уровне токенов, так и на уровне диалога. <sup>†</sup> эти аугментации были реализованы с использованием общедоступных весов модели DSE [43], в то время как в остальных случаях обрезка и перемешивание осуществлялась с помощью вспомогательной модели, обученной на задачу предсказания следующей фразы в диалоге.

Для уточнения деталей реализации аугментаций, обращайтесь к appendixу 1.

## 4.3. Дообучение кодировщика

В качестве кодировщика мы используем модели типа BERT без каких-либо модификаций [21]. На вход подается весь диалог, а на выходе — скрытое

состояние токена [CLS] из последнего слоя либо усредненное скрытое состояние всех токенов. Конкретный выбор процедуры формирования векторного представления из последних скрытых состояний зависит от инициализации весов.

При контрастной донастройке мы получаем положительную пару с помощью аугментаций одного и того же диалога, а остальные диалоги из обучающего батча используются в качестве отрицательных примеров.

Разработанные аугментации отражают некоторые инварианты, присущие диалоговым данным. Поэтому справедливо предположить, что модель, адаптированная под работу с диалогом, не должна выдавать существенно отличающиеся векторные представления объектов до и после аугментаций. Это служит некоторым обоснованием использования контрастивного обучения на аугментациях диалогов как процедуры адаптации языковой модели под домен диалоговых данных.

#### 4.4. Оценивание

Мы оцениваем векторизацию диалогов в контексте задачи переноса знаний (transfer learning). В частности, наши методы оценки используют замороженные векторные представления как признаки, что можно рассматривать как метод пробинга, от англ. «probe» — «зонд». Вдохновленные методами оценки, представленными в [37], мы используем следующие: классификация домена, категоризация диалога, поиск диалога. Также мы используем предсказание намерений как классическую задачу, связанную с задачеориентированными диалогами.

**Классификация домена.** Цель состоит в предсказании того, в какой области сферы услуг происходит диалог. Например, в датасете MultiWOZ 2.2 [44] содержатся 7 областей: достопримечательности, автобус, больница, отель, ресторан, такси, поезд. Метод оценивания заключается в обучении линейного классификатора на векторизации диалога. Этот метод оценки демонстрирует, насколько хорошо кодировщик сохраняет латентные признаки о диалоге. В качестве метрики используется точность.

**Категоризация диалога.** Можно рассматривать классификацию домена как задачу кластеризации и измерять ее качество с помощью метрики чистоты (purity) [45].

**Диалоговый поиск.** Для каждого диалога из валидационного набора данных необходимо найти диалоги из обучающего набора данных с тем же доменом. Меры близости, по которым осуществляется поиск, вычисляются как косинусная близость между двумя диалогами. В качестве метрики используется nDCG@100 [46].

**Предсказание намерений.** По заданному высказыванию необходимо классифицировать намерение. В качестве метрики используется доля верных классификаций (accuracy).

## 5. Эксперименты

### 5.1. Набор данных

Большой набор данных важен для контрастивного обучения. Во всех экспериментах мы использовали один и тот же набор диалогов. Он является комбинацией нескольких открытых коллекций задачеориентированных диалогов [47], которые перечислены в таблице 2. Все диалоги были отфильтрованы на основе их длины, что привело к набору данных, состоящему примерно из 433 тыс. диалогов.

	# диалогов	# реплик	# токенов
AirDialogue	288K	364K	37.3M
SimJointGEN	84K	1122K	15.9M
MS-DC	9K	63K	1M
MetaLWOZ	34K	350K	3.3M
KETOD	2K	36K	0.5M
FRAMES	1K	12K	0.2M
Disambiguation	7K	87K	2M
ABCD	6K	81K	1.1M
Taskmaster1	3K	63K	0.7M
SGD <sup>†</sup>	1K	18K	0.2M
<b>Total</b>	<b>433K</b>	<b>5458K</b>	<b>61.3M</b>

Таблица 2: Все наборы данных взяты из коллекции DialogStudio [47]. Для подсчета столбца # токенов использовался токенизатор BERT. <sup>†</sup>данный набор использовался только для валидации, отфильтрованы диалоги с более чем одним доменом.

### 5.2. Оптимизация набора аугментаций

Сначала мы оценили предварительно обученные модели BERT [21], RetroMAE [31] на открытых наборах данных SGD [48] и banking77 [49]. Результаты представлены в первой строке таблицы 3. Это результаты, которые можно получить с использованием доступных весов без какой-либо дополнительной настройки на домене диалоговых данных. Векторизации, полученные с помощью RetroMAE, имеют значительно более высокие оценки, чем BERT. Это согласуется с высокими оценками RetroMAE на самых разнообразных бенчмарках, включая те, в которых нет диалоговой специфики [10].

**Обучение.** В нашем подходе контрастивного дообучения мы использовали следующие параметры: размер батча 128 на двух A100 40GB, оптимизатор AdamW с гиперпараметром weight\_decay 1e-2 и фиксированный шаг обучения 3e-6. Мы заморозили все слои трансформера, кроме последних трех. Оценки качества после пяти эпох обучения представлены в таблице 3.

Набор	классиф.		кластер.		поиск		намерения	
	bert	retro	bert	retro	bert	retro	bert	retro
raw	65,83	<b>80,46</b>	33,09	<b>89,21</b>	44,32	<b>78,02</b>	23,41	<b>60,29</b>
t-l	73,12	77,93	49,80	60,61	46,31	54,27	30,81	57,18
t-h	77,19	77,65	60,61	69,04	59,45	61,25	34,90	61,33
a-h	76,46	78,49	73,85	78,78	68,31	68,87	45,03	61,75
a-l-m	<b>78,89</b>	79,74	76,80	80,36	<b>71,65</b>	74,28	<b>44,71</b>	47,79
a-h-m	<b>78,89</b>	78,16	<b>77,82</b>	74,87	71,45	70,59	44,45	48,80
a-l-m-dse	78,16	79,91	73,29	80,93	70,30	73,76	44,64	47,82

Таблица 3: Оценка доступных моделей, как описано в разделе 4.4. Для расшифровки названий наборов аугментации обратитесь к разделу 4.2. Качество классификации, кластеризации и поиска измеряется на наборе данных SGD, качество предсказания намерений измеряется на наборе данных banking77. В качестве инициализации использовались две модели: BERT и RetroMAE. Чем выше числа, тем лучше качество.

Согласно результатам, контрастивное дообучение с аугментациями дает прирост качества для оригинальных весов BERT. Наименьший прирост дает набор простейших аугментаций на уровне токенов: замена и вставка. Наилучший прирост дали наборы аугментаций a-l-m и a-h-m. Набор a-l-m-dse имеет наименьшую сложность в реализации и не требует обучения вспомогательной модели. При этом для BERT итоговые метрики имеют большой разрыв с метриками у тривиальных аугментаций, и малый разрыв с метриками у тяжелых аугментаций a-h-m. Поэтому в следующих экспериментах используется набор аугментаций a-l-m-dse как компромиссный между качеством и сложностью реализации.

Контрастивное дообучение с аугментациями не дает прироста качества для инициализации из весов RetroMAE. Для всех сценариев оценивания метрики падают.

### 5.3. Анализ разброса

Был проведён анализ способностей моделей устойчиво векторизовать диалог. На рисунке 1а показана PCA-проекция векторных представлений диалогов из датасета SGD, полученных с помощью BERT. Диалоги из одного домена имеют близкие проекции, однако границы доменов часто пересекаются, некоторые домены «разбросаны» по векторному пространству.

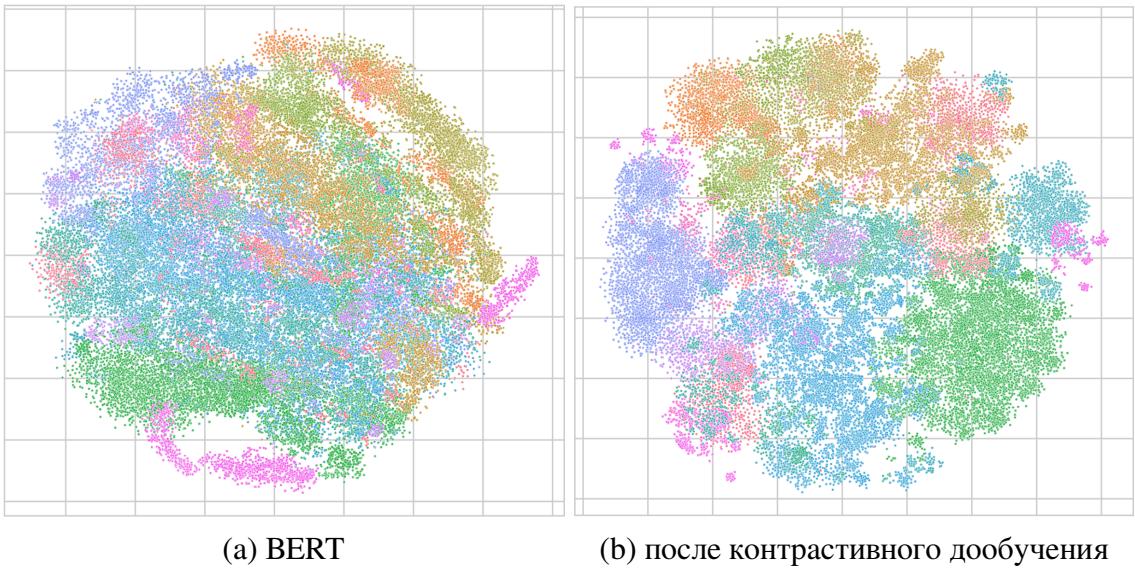


Рис. 1: Проекция векторных представлений диалогов из коллекции SGD, полученная методом главных компонент. Использована модель BERT. Цветом выделена принадлежность диалога к одному из доменов.

На рисунке 1б представлены те же диалоги, но векторизованные уже с помощью модели BERT после процедуры контрастивного дообучения на аугментациях. Облака доменов приняли более округлые формы, в меньшей степени пересекаются и накладываются друг на друга, как это было на рисунке 1а. Веса модели были дообучены без явной размеченной информации о доменах диалогов, при этом по рисунку облака некоторых доменов все равно отчетливо разделимы.

Если провести визуальный анализ таких же графиков для RetroMAE, то оказывается, что его векторные представления изначально хорошо адаптированы под диалоговые данные (рис. 2). Стоит отметить сходство рис. 1б и 2а, поскольку модели BERT с контрастивным дообучением и RetroMAE без контрастивного дообучения были получены совершенно разными путями.

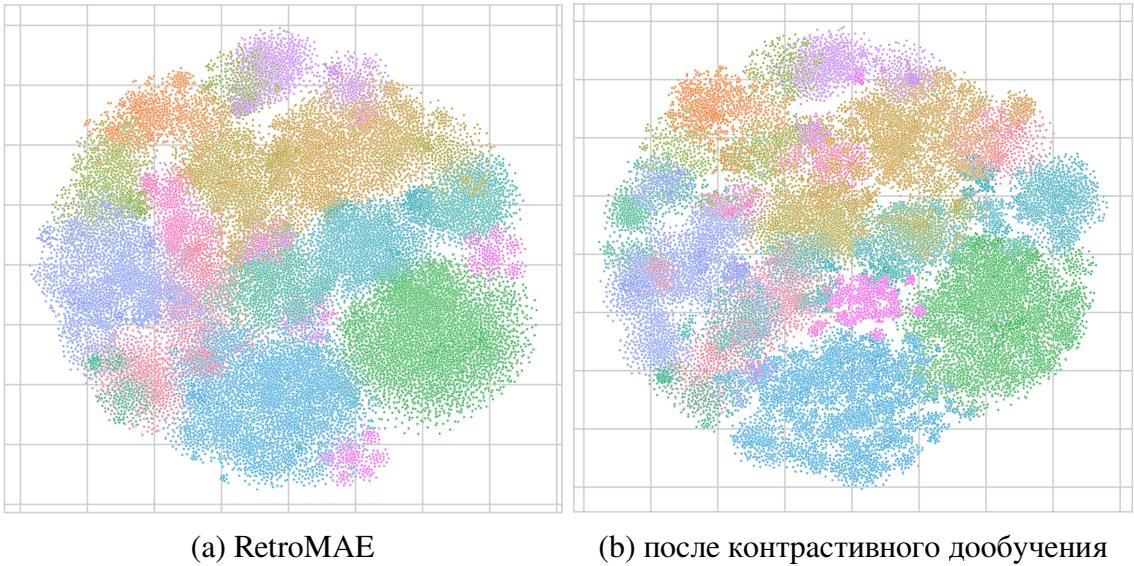


Рис. 2: Проекция векторных представлений диалогов из коллекции SGD, полученная методом главных компонент. Использована модель RetroMAE. Цветом выделена принадлежность диалога к одному из доменов.

#### 5.4. Анализ инвариантности

model	logreg		5nn		cosine	
	raw	cft	raw	cft	raw	cft
BERT	81,41	78,45	40,71	32,16	<b>93,35</b>	62,58
RetroMAE	77,03	75,55	29,05	28,90	87,7	63,92
DSE	80,85	77,17	33,00	30,95	82,38	<b>71,05</b>
BGE	<b>73,64</b>	<b>70,95</b>	<b>25,80</b>	<b>25,37</b>	87,03	68,54

Таблица 4: Результаты классификации диалогов на 5 классов: без аугментации, вставка, замена, обрезка, перемешивание. Для оценки использовалась доля верных классификаций (accuracy). «raw» означает исходные веса, «cft» означает контрастивно дообученную модель на наборе аугментаций a-l-m-dse. Чем ниже значение, тем лучше, поскольку это показывает инвариантность к примененным аугментациям входных данных.

Исследована способность моделей улавливать инварианты диалогов. Для этого для каждой модели векторизации обучены несколько более простых моделей на задачу классификации, в которой меткой является то, какая аугментация была произведена над диалогом, в том числе отсутствие аугментаций. Результаты представлены в таблице 4. После дообучения при различных инициализациях предсказать тип аугментации удается с меньшей точностью, чему могут послужить две невзаимоисключающие причины: 1) модель приобретает инвариантность к аугментациям, что говорит о её адап-

тированности к диалоговым данным, 2) общие способности этой модели понимать язык ухудшились. Однако вторая причина не согласуется с тем, что более сильные кодировщики вроде DSE [31] и BGE [7] имеют еще более низкое качество предсказания аугментации, особенно метрическими методами. Дополнительно стоит отметить, что чем сильнее кодировщик в целом для текстов [10], тем меньше он показывает точность предсказания аугментации.

## 5.5. Анализ скрытых состояний

Дополнительно мы исследовали качество предсказания аугментации с использованием векторного представления, взятого на более глубоких слоях. Результаты представлены в таблице 5. Оказалось, что качество предсказания резко падает на последнем слое, особенно после контрастивного дообучения на аугментациях. Другими словами, последние слои стараются получить представления, более инвариантные к аугментациям и как следствие, адаптировать модель под данные.

model	raw				cft			
	emb	hs[-2]	hs[-3]	hs[-4]	emb	hs[-2]	hs[-3]	hs[-4] <sup>†</sup>
BERT	40,71	43,46	42,12	43,18	32,16	41,06	43,96	43,18
RetroMAE	29,05	38,59	38,37	39,29	28,90	<b>37,95</b>	39,22	39,29
DSE	33,00	37,53	38,73	41,20	30,95	39,58	43,53	41,20
BGE	<b>25,80</b>	<b>31,73</b>	<b>33,99</b>	<b>38,52</b>	<b>25,37</b>	41,91	<b>38,94</b>	<b>38,52</b>

Таблица 5: Результаты классификации диалогов с помощью метода пяти ближайших соседей на 5 классов: без аугментации, вставка, замена, обрезка, перемешивание. Для оценки использовалась доля верных классификаций (accuracy). «raw» означает исходные веса, «cft» означает контрастивно дообученную модель на наборе аугментаций a-l-m-dse. <sup>†</sup>Качество предсказания не меняется после дообучения, поскольку эти слои были «заморожены».

## 6. Заключение

Было проведено исследование нейросетевых векторных представлений для задачеориентированных диалогов. Предложена процедура контрастивного обучения с аугментациями. Эффективность этой процедуры была продемонстрирована для BERT-подобных моделей в задачах, таких как классификация диалога, кластеризация диалога, поиск диалога, классификация реплик. Было проведено аблационное исследование, показывающее вклад разных аугментаций в качество итоговой модели. Вместе с дополнительными исследованиями инвариантности кодировщиков к аугментациям можно сделать следующие выводы:

- доступные сегодня кодировщики общего назначения достаточно хорошо адаптированы к диалоговым данным;
- контрастивное обучение с аугментациями достаточно дешевая процедура для получения эмбедингов, адаптированных под заданный домен данных.

Дальнейшие исследования могут быть связаны с более тонкой настройкой процедуры обучения с точки зрения подбора гиперпараметров, поскольку в данной работе не удалось превзойти качество существующих моделей.

# Список литературы

1. *Qader W. A., Ameen M. M., Ahmed B. I.* An Overview of Bag of Words; Importance, Implementation, Applications, and Challenges // 2019 International Engineering Conference (IEC). — 2019. — С. 200—204. — DOI: 10.1109/IEC47844.2019.8950616.
2. *Jones K. S.* A statistical interpretation of term specificity and its application in retrieval // J. Documentation. — 2021. — Т. 60. — С. 493—502. — URL: <https://api.semanticscholar.org/CorpusID:2996187>.
3. Efficient Estimation of Word Representations in Vector Space / T. Mikolov [и др.]. — 2013.
4. *Pennington J., Socher R., Manning C.* GloVe: Global Vectors for Word Representation // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). — Doha, Qatar : Association for Computational Linguistics, 10.2014. — С. 1532—1543. — DOI: 10.3115/v1/D14-1162. — URL: <https://aclanthology.org/D14-1162>.
5. Learned in Translation: Contextualized Word Vectors / B. McCann [и др.]. — 2018.
6. Deep Contextualized Word Representations / M. E. Peters [и др.] // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). — New Orleans, Louisiana : Association for Computational Linguistics, 06.2018. — С. 2227—2237. — DOI: 10.18653/v1/N18-1202. — URL: <https://aclanthology.org/N18-1202>.
7. C-Pack: Packaged Resources To Advance General Chinese Embedding / S. Xiao [и др.]. — 2023.

8. Text Embeddings by Weakly-Supervised Contrastive Pre-training / L. Wang [и др.]. — 2022.
9. Towards General Text Embeddings with Multi-stage Contrastive Learning / Z. Li [и др.]. — 2023.
10. MTEB: Massive Text Embedding Benchmark / N. Muennighoff [и др.]. — 2023.
11. *Oord A. van den, Li Y., Vinyals O.* Representation Learning with Contrastive Predictive Coding. — 2019.
12. Dialog-Post: Multi-Level Self-Supervised Objectives and Hierarchical Model for Dialogue Post-Training / Z. Zhang [и др.] // Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Toronto, Canada : Association for Computational Linguistics, 07.2023. — C. 10134—10148. — DOI: 10.18653/v1/2023.acl-long.564. — URL: <https://aclanthology.org/2023.acl-long.564>.
13. *Li Y., Zhao H., Zhang Z.* Back to the Future: Bidirectional Information Decoupling Network for Multi-turn Dialogue Modeling. — 2022.
14. *Kim S., Chang M., Lee S.-W.* NeuralWOZ: Learning to Collect Task-Oriented Dialogue via Model-Based Simulation. — 2021.
15. Simulated Chats for Building Dialog Systems: Learning to Generate Conversations from Instructions / B. Mohapatra [и др.]. — 2021.
16. A Unified Dialogue User Simulator for Few-shot Data Augmentation / D. Wan [и др.] // Findings of the Association for Computational Linguistics: EMNLP 2022. — Abu Dhabi, United Arab Emirates : Association for Computational Linguistics, 12.2022. — C. 3788—3799. — DOI: 10.18653/v1/2022.findings-emnlp.277. — URL: <https://aclanthology.org/2022.findings-emnlp.277>.
17. AugESC: Dialogue Augmentation with Large Language Models for Emotional Support Conversation / C. Zheng [и др.]. — 2023.
18. *Schick T., Schütze H.* Generating Datasets with Pretrained Language Models. — 2021.

19. *Soudani H., Kanoulas E., Hasibi F.* Data Augmentation for Conversational AI // arXiv preprint arXiv:2309.04739. — 2023.
20. *Reimers N., Gurevych I.* Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. — 2019.
21. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / J. Devlin [и др.]. — 2019.
22. *Noroozi M., Favaro P.* Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles. — 2017.
23. *Doersch C., Zisserman A.* Multi-task Self-Supervised Visual Learning. — 2017.
24. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination / Z. Wu [и др.]. — 2018.
25. Text and Code Embeddings by Contrastive Pre-Training / A. Neelakantan [и др.]. — 2022.
26. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval / L. Xiong [и др.]. — 2020.
27. Language-agnostic BERT Sentence Embedding / F. Feng [и др.]. — 2022.
28. *Williams A., Nangia N., Bowman S.* A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). — New Orleans, Louisiana : Association for Computational Linguistics, 2018. — C. 1112—1122. — URL: <http://aclweb.org/anthology/N18-1101>.
29. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering / Z. Yang [и др.] // Conference on Empirical Methods in Natural Language Processing (EMNLP). — 2018.
30. *Kotonya N., Toni F.* Explainable Automated Fact-Checking for Public Health Claims // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). — Online : Association for Computational Linguistics, 11.2020. — C. 7740—7754. — URL: <https://www.aclweb.org/anthology/2020.emnlp-main.623>.

31. RetroMAE: Pre-Training Retrieval-oriented Language Models Via Masked Auto-Encoder / S. Xiao [и др.]. — 2022.
32. *Gao L., Callan J.* Condenser: a Pre-training Architecture for Dense Retrieval // Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing / под ред. M.-F. Moens [и др.]. — Online, Punta Cana, Dominican Republic : Association for Computational Linguistics, 11.2021. — C. 981—993. — DOI: 10.18653/v1/2021.emnlp-main.75. — URL: <https://aclanthology.org/2021.emnlp-main.75>.
33. CERT: Contrastive Self-supervised Learning for Language Understanding / H. Fang [и др.]. — 2020.
34. ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer / Y. Yan [и др.] // Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) / под ред. C. Zong [и др.]. — Online : Association for Computational Linguistics, 08.2021. — C. 5065—5075. — DOI: 10.18653/v1/2021.acl-long.393. — URL: <https://aclanthology.org/2021.acl-long.393>.
35. Unsupervised Document Embedding via Contrastive Augmentation / D. Luo [и др.]. — 2021.
36. *Miller G. A.* WordNet: a lexical database for English // Commun. ACM. — New York, NY, USA, 1995. — Нояб. — Т. 38, № 11. — С. 39—41. — DOI: 10.1145/219717.219748. — URL: <https://doi.org/10.1145/219717.219748>.
37. Dial2vec: Self-Guided Contrastive Learning of Unsupervised Dialogue Embeddings / C. Liu [и др.]. — 2022.
38. DialogueCSE: Dialogue-based Contrastive Learning of Sentence Embeddings / C. Liu [и др.]. — 2021.
39. RoBERTa: A Robustly Optimized BERT Pretraining Approach / Y. Liu [и др.]. — 2019.
40. Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation / Y. Wu [и др.]. — 2016.

41. *Sennrich R., Haddow B., Birch A.* Neural Machine Translation of Rare Words with Subword Units // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Berlin, Germany : Association for Computational Linguistics, 08.2016. — C. 1715—1725. — DOI: 10 . 18653 / v1 / P16 - 1162. — URL: <https://aclanthology.org/P16-1162>.
42. *Tiedemann J., Thottingal S.* OPUS-MT — Building open translation services for the World // Proceedings of the 22nd Annual Conference of the European Association for Machine Translation (EAMT). — Lisbon, Portugal, 2020.
43. Learning dialogue representations from consecutive utterances / Z. Zhou [и др.] // NAACL 2022. — 2022. — URL: <https://www.amazon.science/publications/learning-dialogue-representations-from-consecutive-utterances>.
44. MultiWOZ 2.2 : A Dialogue Dataset with Additional Annotation Corrections and State Tracking Baselines / X. Zang [и др.]. — 2020.
45. *Palacio-Niño J.-O., Berzal F.* Evaluation Metrics for Unsupervised Learning Algorithms. — 2019.
46. A Theoretical Analysis of NDCG Type Ranking Measures / Y. Wang [и др.]. — 2013.
47. DialogStudio: Towards Richest and Most Diverse Unified Dataset Collection for Conversational AI / J. Zhang [и др.] // Findings of the Association for Computational Linguistics: EACL 2024 / под ред. Y. Graham, M. Purver. — St. Julian's, Malta : Association for Computational Linguistics, 03.2024. — C. 2299—2315. — URL: <https://aclanthology.org/2024.findings-eacl.152>.
48. Towards Scalable Multi-domain Conversational Agents: The Schema-Guided Dialogue Dataset / A. Rastogi [и др.]. — 2020.
49. Efficient Intent Detection with Dual Sentence Encoders / I. Casanueva [и др.]. — 2020.
50. HuggingFace’s Transformers: State-of-the-art Natural Language Processing / T. Wolf [и др.]. — 2020.

51. ConveRT: Efficient and Accurate Conversational Representations from Transformers / M. Henderson [и др.]. — 2020.

# 1. Вспомогательные модели

Как упомянуто в разделе 4.1, мы обучали вспомогательные модели для реализации аугментаций обрезка и перемешивание.

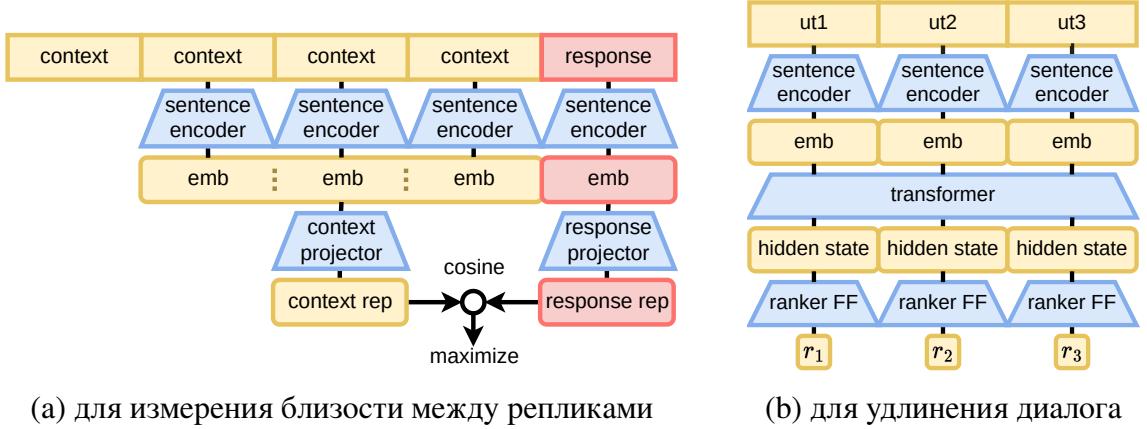


Рис. 3: Вспомогательные модели для реализации аугментаций.

## 1.1. Близость между репликами в диалоге

Мы обучили модель для измерения сходства между высказываниями в диалоге. В простейшем виде её можно реализовать следующим образом: взять векторы высказываний и посчитать косинусы между ними. Обучение такой модели на последовательных репликах из одного диалога с использованием контрастного обучения дает богатую векторизацию [43]. Однако у этого подхода есть существенный недостаток: он не учитывает контекст из нескольких предшествующих высказываний. В диалоге важно сравнивать не только пары высказываний, но и пары контекст-ответ. Поэтому мы использовали следующую модель (рис. 3а):

- 1) Сначала векторные представления для всех контекстных высказываний  $c = [u_1, \dots, u_k]$  и ответа  $r$  получаются с использованием предварительно обученного кодировщика.
- 2) Векторы контекста конкатенируются и передаются на проектор, который выдает вектор, представляющий контекст.
- 3) Вектор ответа подается на второй проектор.

- 4) Косинусная близость между полученными векторами вычисляется как мера близости контекста и ответа.

Модель `aws-ai/dse-bert-large` от Hugging Face [50] и Amazon [43] использовалась в качестве кодировщика. Модель обучалась с использованием контрастивной функции потерь с отрицательными примерами из батча, с следующими параметрами: размер пакета 128, температура 0.05, размер контекста 3, размер проекции 256. Только последние 3 слоя кодировщика предложений были дообучены для снижения вычислительной стоимости. Обученная модель достигает точности 0.955 accuracy@5 внутри батча.

Пары "контекст-ответ" для формирования батчей были взяты из всего набора данных диалогов, где отрицательные примеры не были выбраны из того же диалога, а представляли собой полностью случайные примеры из набора данных.

Полученная модель напоминает модель ConveRT [51] для получения векторных представлений высказываний. Недостатки последней модели заключаются, во-первых, в её закрытости, и, во-вторых, в том, что её архитектура является высокоспециализированной и не использует повсеместно используемый компонент типа BERT.

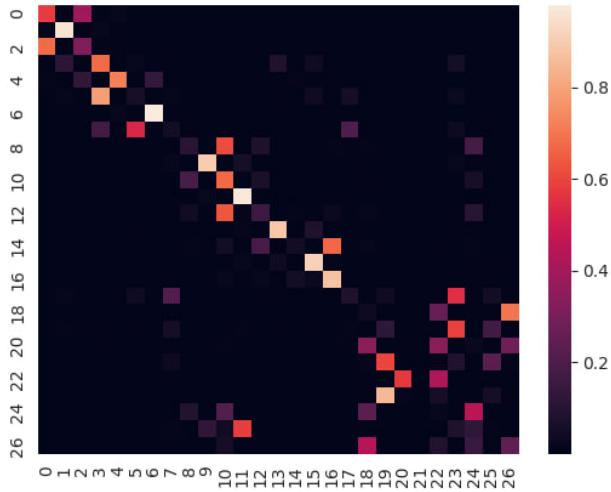


Рис. 4: Все сходства между контекстами и ответами в рамках диалога. Легко видеть, что последовательные реплики образуют кластеры.

В результате полученная модель способна распознавать отдельные этапы в диалогах (рис. 4). Это поведение достигается благодаря двум факторам. Во-первых, когда диалог переходит к новой теме, сходство между двумя последовательными высказываниями существенно снижается. Это явно вид-

но, например, в диалогах, где заказ такси сменяет бронирование столика в ресторане. Во-вторых, в рамках одной темы также наблюдаются небольшие падения сходства в случаях, когда происходит переход от одного вопроса к другому. Например, вопрос «на сколько человек мне нужно забронировать?» сменяет вопрос «в каком ресторане вы предпочитаете?». Более того, поскольку эти вопросы относятся к одной теме, они остаются близкими друг к другу и далекими от вопросов по другим темам. Поэтому образуются кластеры.

В экспериментах также фигурируют аугментации обрезка и перемешивание без использования вспомогательной модели и окна контекста. Вместо не используется один лишь претренированный кодировщик DSE [43]. Это сделано для того, чтобы показать, что достойный результат можно достичь и с использованием открытых моделей без дообучения.

## 1.2. Списочная модель

Мы рассматривали ещё одну вспомогательную модель для аугментации, которую не включили в основную часть работы. Мы обучили специальную модель для объединения высказываний из двух разных диалогов (рис. 3b). Это трансформер над векторными представлениями реплик. Мы использовали 4-слойный трансформер с 4 головами внимания и скрытой размерностью вдвое меньше, чем у кодировщика предложений, т.е. 384. Только последние 3 слоя кодировщика предложений были дообучены для снижения вычислительных затрат.

Ранги на выходе преобразуются с помощью функции softmax. Затем минимизируется дивергенция Кульбака-Лейблера между выводом и целевыми вероятностями. Целевые вероятности определяются как softmax по истинным рангам высказываний, т.е.  $-i$  для  $i$ -го высказывания в диалоге.

Полученная модель обучается "сортировать" высказывания согласно тому, как бы они гипотетически стояли в диалоге. Благодаря механизму внимания трансформеров это можно рассматривать как просьбу детям на уроке физической культуры смотреть друг на друга и выстраиваться по росту.

Для измерения качества сортировки нам необходимо использовать соответствующую метрику. Все традиционные метрики ранжирования, такие как nDCG, разработаны для сравнения с золотыми рангами, а не только для оценки качества сортировки. Поэтому во время валидации нашей модели мы

преобразовывали ранги в перестановку исходной последовательности из  $n$  элементов. Затем мы вычисляли количество транспозиций. Это легко реализовать и может быть нормализовано по максимально возможному числу транспозиций  $n(n - 1)/2$ , что приводит к метрике в диапазоне  $[0, 1]$ . Наша обученная модель достигает значения 0.96.

## 2. Композиции аугментаций

Для максимизации разнообразия обучающих данных мы используем не только 5 основных аугментаций, описанных в разделе 4.1, но и 4 дополнительные комбинации аугментаций. Все полученные конвейеры определены на рис. 5.

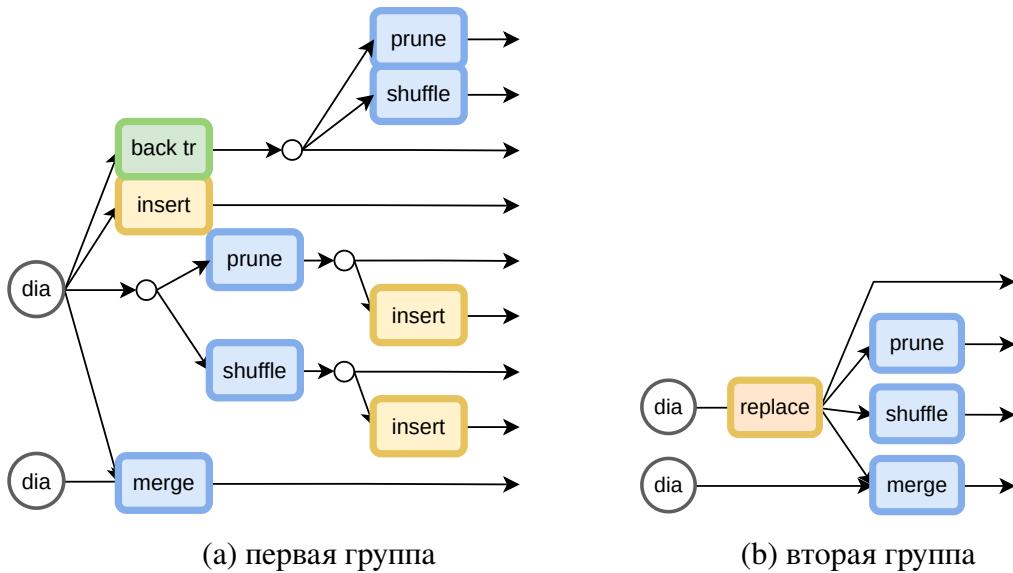


Рис. 5: Композиции аугментаций.