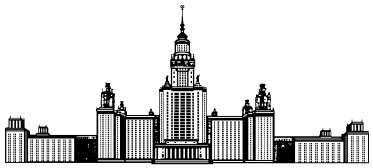

Московский государственный университет имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра математических методов прогнозирования

Сенин Александр Николаевич

Методы реализации расстояний с использованием разнородной обучающей информации

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Научный руководитель:
к.ф.-м.н.
Майсурадзе Арчил Иериевич

Москва, 2021

Содержание

Аннотация

Данная работа посвящена задаче реализации расстояний, в частности новому подходу к методам многомерного шкалирования. Для основных методов многомерного шкалирования, а именно метрического и неметрического, предложена модель машинного обучения (МО), их реализующая. Сравнение с известными алгоритмами показало преимущество предложенной модели.

Проведен обзор методов оценки качества векторных представлений слов. По результатам обзора поставлена задача построения векторных представлений слов на основе разнородной обучающей информации, причем в этой роли взяты наборы данных для оценивания качества векторных представлений слов: размеченные экспертами степени семантической близости и наборы аналогий. На основе ранее полученной модели МО для задач многомерного шкалирования создана модель МО для задачи построения векторных представлений слов. Сравнительный анализ обучающих наборов позволил выбрать наилучшие для обучения. При этом качество полученных представлений сопоставимо с представлениями, полученными SOTA-подходами.

Реализована программная библиотека работы с наборами семантической близости и наборами аналогий, автоматизирующая оценку качества векторных представлений слов. Реализованы все исследованные модели МО на современных платформах оптимизации вычислительных графов.

1 Введение

В задачах машинного обучения и анализа данных актуальными являются задачи понижения размерности и визуализации данных. Понижение размерности данных позволяет «бороться» с переобучением, снижать вычислительные затраты, сжимать объемы хранимых и используемых данных. Визуализация данных, в свою очередь, дает возможность представить данные некоторой аудитории, провести исследовательский анализ данных, обнаружить в данных закономерности или дефекты.

Все эти задачи тесно связаны с задачами реализации расстояний. Под задачами реализации расстояний неформально понимается класс задач, где требуется по заданным расстояниям найти конфигурацию объектов (объектами могут выступать вершины в графе или, например, точки в евклидовом пространстве, рассмотренные в этой работе). Методы многомерного шкалирования позволяют решать эту задачу. Общая идея таких методов заключается в нахождении конфигурации точек в евклидовом пространстве небольшой размерности по заданным различиям на парах объектов. В дальнейшем эти точки могут рассматриваться как признаковые представления объектов и использоваться в последующих моделях или для визуализации.

Стоит отметить, что перечень применений не ограничен понижением размерности и визаулизацией. Задача понижения размерности подразумевает существование признакового описания объектов в данных некоторой большой размерности. Методы многомерного шкалирования в свою очередь не требуют существования какого-либо признакового представления данных, достаточно лишь различий на парах объектов. Эта важная особенность позволяет применять эти методы для построения векторных представлений объектов самой разной природы, достаточно лишь иметь функцию, возвращающую некоторое число на паре объектов, несущее смысл различия. Эта особенность будет активно использоваться в данной работе для решения задачи построения векторных представлений слов. Существуют размеченные экспертами степени семантической близости на парах слов, которые можно преобразовать в различия. Помимо этого, распространение получил метод оценивания векторных представлений слов на основе наборов аналогий.

Таким образом, цель данной работы — предложить новый подход к решению задач многомерного шкалированя, а также новый способ построения векторных представлений слов на основе разнородной обучающей информации (семантическая близость и аналогия) методами многомерного шкалирования. Соответственно, задачи

данной работы: разработать и реализовать модель МО, решающую задачи метрического и неметрического многомерного шкалирования, реализовать программную библиотеку для работы с разнородными исходными данными, разработать модель МО для построения векторных представлений слов методами многомерного шкалирования на основе разнородных данных.

2 Постановка задачи

2.1 Обзор литературы

Задачи реализации расстояний — общее название обширного класса задач. Такие задачи возникают в самых разных предметных областях, имеют разнообразные формулировки и методы решений. Чаще всего, задачи этого класса сводятся к нахождению конфигурации объектов, которая наилучшим образом приближает заданные расстояния. Объектами могут выступать вершины в графе, а конечное применение задача может находить, например, в моделировании трафика в интернете [4]. Не менее часто в роли объектов выступают точки евклидова пространства, а входные расстояния задаются взвешенным графом. Такие задачи называют задачами реализации графов, они возникают в биологии, статистике, робототехнике [7]. Наконец, класс задач многомерного шкалирования тоже входит в класс задач реализации расстояний. К нему относятся задачи классического многомерного шкалирования [5], метрического и неметрического [10], обобщенного многомерного шкалирования [1]. Эти задачи находят применение в психологии, экономике, статистике. Отдельно рассмотрим класс таких задач, для этого введем основные понятия.

2.2 Основные понятия

Пусть $\mathcal{X}$ — множество объектов произвольной природы.

Метрикой (*расстоянием*) будем называть функцию $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $\forall x, y, z \in \mathcal{X}$ удовлетворяющую следующим аксиомам:

(a) $\rho(x, x) = 0$

(b) $\rho(x, y) \geqslant 0$ (неотрицательность)

(c) $\rho(x, y) = \rho(y, x)$ (симметричность)

(d) $\rho(x, y) = 0 \Longrightarrow x = y$ (определенность)

(e) $\rho(x, y) \leqslant \rho(x, z) + \rho(z, y)$ (неравенство треугольника)

Легко показать, что аксиома неотрицательности (c) является избыточной и вытекает из оставшихся аксиом:

$$0 = \rho(x, x) \leqslant \rho(x, y) + \rho(y, x) = 2\rho(x, y) \Longrightarrow \rho(x, y) \geqslant 0$$

Тем не менее, кажется интуитивным ожидать неотрицательности от понятия расстояния, поэтому принято оставлять эту аксиому. Следуя за [12], определим функцию различия (dissimilarity function). Другое наименование — функция расстояния (distance function).

Различием (*dissimilarity*) будем называть функцию $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $\forall x, y \in \mathcal{X}$ удовлетворяющую следующим требованиям:

(D1) $d(x, x) = 0$

(D2) $d(x, y) \geqslant 0$

Эти требования кажутся легко выполнимыми для любой функции, несущей смысл различия на паре объектов: различие должно быть неотрицательным, кроме того, объект ничем не должен различаться от себя самого. Разумеется, неформально ожидается сохранение смысла различия при построении такой функции. Тождественный ноль, строго говоря, тоже является различием, но не несет в себе содержательного смысла. Интуитивная логика работы различия: чем ближе между собой объекты в терминах близости предметной области множества $\mathcal{X}$, тем меньше значение функции различия.

Различие будем считать *обобщением* метрики. Помимо различия существуют и другие обобщения: квазиметрика, псевдометрика, полуметрика, метаметрика и другие. Эти обобщения накладывают дополнительные требования на функцию различия, удаляя по одной или несколько аксиом из системы аксиом метрики.

Рассмотрим теперь некоторое множество объектов $x_1, ..., x_N \in \mathcal{X}$. Известно, что на некоторых парах этого множества определена функция различия $d(x_i, x_j)$. В таком случае заданы тройки $\{(i, j, d_{ij})\}_{(i,j)\in M}$, где M — множество пар объектов, на которых определено различие. В случае $M = (i, j)_{i,j=1}^{N}$ будем считать, что задана *матрица*

попарных различий $D = (d_{ij})_{i,j=1}^{N}$. В случае, когда известно, что d представляет собой метрику, задана *матрица попарных расстояний* D.

Интересной особенностью функций различия является возможность преобразовать ее в полноценную метрику. Способы такого преобразования описаны в [12].

Кажется естественным вслед за различием определить понятие сходства. Здесь мы вновь последуем за [12].

Сходством (*similarity*) будем называть функцию $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $\forall x \in \mathcal{X}$ удовлетворяющую единственному требованию:

(S1) $s(x, x) > 0$

Аналогично различию, мы неформально ожидаем от функции сходства такой логики работы: чем ближе между собой объекты в терминах близости предметной области множества $\mathcal{X}$, тем больше значение функции сходства.

Кажется логичным дополнительно потребовать симметричность и неотрицательность от функции сходства. Однако, не всегда на практике выполняются эти свойства. Пример несимметричного сходства мы встретим далее в работе, когда пойдет речь о сходствах на парах слов. В качестве функции сходства между двумя векторами часто используют косинус угла между x и y или скалярное произведение $\langle x, y \rangle$, применение таких функций мы так же встретим далее в работе. В обоих случаях функция сходства может принимать отрицательные значения.

Будем говорить, что для троек $\{(i, j, d_{ij})\}_{(i,j)\in M}$ существует *реализация расстояния* (*различия*) в метрическом пространстве $(\mathbb{R}^p, \rho)$, если существует такой набор (*конфигурация*) $x_1, ..., x_N \in \mathbb{R}^p$, что $\forall (i, j) \in M$

$$\rho(x_i, x_j) = d_{ij} \tag{1}$$

На практике зачастую при некотором фиксированном p не удается найти реализации расстояния, то есть конфигурации, на которой выполняется точное равенство. Здесь мы приходим к основному подходу методов многомерного шкалирования — поиск конфигурации, которая лучше всего приближает $\rho(x_i, x_j) \approx d_{ij}$.

В литературе часто конкретизируют вид метрики ρ, задавая ее евклидовым расстоянием. Тем не менее, рассмотренные подходы работают и с произвольной метрикой, если это не оговорено отдельно. Аналогично, возможно использование не метрики, а некоторой функции различия d. Все понятия легко переносятся на эти случаи.

2.3 Преобразования сходств и различий

Часто приходится конвертировать сходства в различия или наоборот, например, в случае, когда алгоритм работает с различиями, а данные заданы в виде сходств (именно этот случай рассмотрен далее в работе).

По сходствам получаем различия:

- Пусть известен диапазон значений функции сходства $0 \leqslant s(x, y) \leqslant S$, $\forall x, y \in \mathcal{X}$, а также $s(x, x) = S$, $\forall x \in \mathcal{X}$ (среди всевозможных объектов больше всего на себя похож сам объект). Тогда легко показать, что $d(x, y) = S - s(x, y)$ является функцией различия.

- Более общий подход заключается в применении некоторой неотрицательной убывающей функции одного аргумента к функции сходства $f(s(x, y))$, где $f(a) > f(b)$ при $a < b$. В общем случае мы можем получать не строгое различие — терять требования (D1) или (D2). Очевидный вариант преобразования $d(x, y) = \dfrac{1}{s(x, y)}$ лишает нас требования (D1). На практике сохранение всех аксиом функции различия зачастую не столь важно, вариация функции f позволяет смещать внимание модели на более тщательное воспроизведение близких (или наоборот далеких) объектов.

По различиям получаем сходства:

- Общий подход абсолютно аналогичен преобразованию сходств в различия: применяем некоторую убывающую положительную функцию к функции различия $g(d(x, y))$, где $g(a) > g(b)$ при $a < b$. Слабые требования в определении функции сходства позволяют гарантировать получение строгого сходства при таком преобразовании.

Эти способы представляют собой эвристики, однако, существуют способы более качественного преобразования сходств и различий с некоторыми теоретическими гарантиями при дополнительных условиях, такие способы описаны в [12].

2.4 Многомерное шкалирование

Рассмотрим теперь некоторое множество объектов $s_1, ..., s_N \in \mathcal{X}$. Известно, что на некоторых парах M этого множества определена функция различия $d(x, y)$ и известны ее значения: $\{(i, j, d_{ij})\}_{(i,j)\in M}$, где $d_{ij} = d(s_i, s_j)$.

Идейно общий подход методов *многомерного шкалирования* (*multidimensional scaling, MDS*) заключается в нахождении (возможно приближенно) конфигурации $x_1, ..., x_N \in \mathbb{R}^p$, реализующей различия d_{ij} в терминах (1).

Принято считать, что размерность p полученных векторов $x_1, ..., x_N$ задается предварительно. Однако, существуют теоретические результаты, гарантирующие существование размерности q и конфигурации $x_1, ..., x_N \in \mathbb{R}^q$, точно реализующей матрицу попарных расстояний D в терминах (1), при условии $d(x, y) = \rho_E(x, y) = ||x-y||$ — исходная функция различия должна быть расстоянием, причем евклидовым.

Вслед за [5] выделим три основных подхода к решению поставленной задачи:

1. *Классическое многомерное шкалирование (classical multidimensional scaling, cMDS):*

   Пусть задана матрица попарных различий $D = (d_{ij})_{i,j=1}^N$ и выполнено важное предположение: $\mathcal{X}$ является евклидовым пространством (линейное векторное пространство с определенным скалярным произведением), причем функция различия d есть евклидово расстояние $d(s_i, s_j) = \rho_E(s_i, s_j) = ||s_i - s_j|| = \sqrt{\langle s_i - s_j, s_i - s_j \rangle}$.

   Обозначим *матрицу Грама* (матрицу скалярных произведений):

   $$B = (b_{ij})_{i,j=1}^N, \text{ где } b_{ij} = \langle s_i, s_j \rangle. \tag{2}$$

   Определим *функционал Strain (натяжения)*:

   $$Strain(x_1, ..., x_N) = \sum_{i,j=1}^{N} (b_{ij} - \langle x_i, x_j \rangle)^2, \ x_1, ..., x_N \in \mathbb{R}^p \tag{3}$$

   Часто определяют $Strain$ иначе:

   $$Strain(x_1, ..., x_N) = \sqrt{\frac{\sum_{i,j=1}^{N} (b_{ij} - \langle x_i, x_j \rangle)^2}{\sum_{i,j=1}^{N} b_{ij}^2}}, \ x_1, ..., x_N \in \mathbb{R}^p \tag{4}$$

   Нормировочный знаменатель в (4) добавляют, чтобы оценить качество восстановленной конфигурации $x_1, ..., x_N$ относительно квадратов отклонения скалярных произведений без учета порядка самих скалярных произведений b_{ij}, корень добавляют, чтобы получить величину в исходных единицах измерения (аналогия со стандартным отклонением в статистике).

   Задача классического многомерного шкалирования:

- По матрице попарных различий D восстановить матрицу Грама B.
- Решить оптимизационную задачу:

$$Strain(x_1, ..., x_N) = \sum_{i,j=1}^{N} (b_{ij} - \langle x_i, x_j \rangle)^2 \longrightarrow \min_{x_1,...,x_N \in \mathbb{R}^p} \tag{5}$$

Легко заметить, что для оптимизационной задачи минимизации определения (3) и (4) эквивалентны.

2. *Метрическое многомерное шкалирование (metric multidimensional scaling, mMDS):*

Этот подход обобщает классический подход, снимая требования на заданную матрицу попарных различий.

Итак, пусть на некоторых парах M заданы значения функции различия $d(x, y)$: $\{(i, j, d_{ij})\}_{(i,j)\in M}$, где $d_{ij} = d(s_i, s_j)$.

Определим *функционал Stress (стресса)*:

$$Stress(x_1, ..., x_N) = \sum_{(i,j)\in M} (d_{ij} - \rho(x_i, x_j))^2,\ x_1, ..., x_N \in \mathbb{R}^p \tag{6}$$

Часто определяют $Stress$ иначе:

$$Stress(x_1, ..., x_N) = \sqrt{\frac{\sum_{(i,j)\in M} (d_{ij} - \rho(x_i, x_j))^2}{\sum_{(i,j)\in M} d_{ij}^2}},\ x_1, ..., x_N \in \mathbb{R}^p \tag{7}$$

Аналогично, нормировочный знаменатель в (7) добавляют, чтобы оценить качество восстановленной конфигурации $x_1, ..., x_N$ относительно квадратов отклонения различий без учета порядка самих различий d_{ij}, корень добавляют, чтобы получить величину в исходных единицах измерения.

Задача метрического многомерного шкалирования:

- Решить оптимизационную задачу:

$$Stress(x_1, ..., x_N) = \sum_{(i,j)\in M} (d_{ij} - \rho(x_i, x_j))^2 \longrightarrow \min_{x_1,...,x_N \in \mathbb{R}^p} \tag{8}$$

Аналогично, легко заметить, что для оптимизационной задачи минимизации определения (6) и (7) эквивалентны.

3. *Неметрическое многомерное шкалирование (non-metric multidimensional scaling, nMDS):*

   В классической литературе, например [10], обычно предполагается существование некоторой неизвестной возрастающей функции f, такой что $d_{ij} = f(\delta_{ij})$, где d_{ij} — наблюдаемые различия, а δ_{ij} — истинные. Затем по аналогии с (6) конструируется функционал, минимизация которого происходит с учетом лишь порядка на наблюдаемых различиях d_{ij}. Этот подход обсудим позднее.

   Мы же, вслед за [1], сформулируем требование сохранения порядка на различиях иначе.

   Пусть на некоторых парах M заданы значения функции различия $d(x, y)$: $\{(i, j, d_{ij})\}_{(i,j)\in M}$, где $d_{ij} = d(s_i, s_j)$.

   Задача неметрического многомерного шкалирования:

   - По заданным различиям d_{ij} требуется найти конфигурацию $x_1, ..., x_N \in \mathbb{R}^p t$ такую, что

   $$\forall\, (i,j),(k,l) \in M \;\; d_{ij} < d_{kl} \Longleftrightarrow \rho(x_i, x_j) < \rho(x_k, x_l) \tag{9}$$

   - Дополнительно можно потребовать выполнение неравенства с некоторым зазором (отступом) $m > 0$:

   $$\forall\, (i,j),(k,l) \in M \;\; d_{ij} < d_{kl} \Longleftrightarrow \rho(x_i, x_j) + m < \rho(x_k, x_l) \tag{10}$$

3 Подходы к решению задач многомерного шкалирования

Обсудим классические подходы к решению рассмотренных задач.

1. *Классическое многомерное шкалирование (classical multidimensional scaling, cMDS):*

   При решении cMDS важны два предположения:

   - Предположение о том, что функция различия есть евклидово расстояние $d(s_i, s_j) = \rho_E(s_i, s_j) = ||s_i - s_j|| = \sqrt{\langle s_i - s_j, s_i - s_j\rangle}$

- Предположение о том, что расстояния заданы на всевозможных парах (i, j), то есть задана полноценная матрица попарных расстояний $D = (d_{ij})_{i,j=1}^N$.

Основная идея заключается в возможности выразить квадрат евклидового расстояния через скалярные произведения:

$d_{ij}^2 = (x_i - x_j)^T(x_i - x_j) = x_i^T x_i + x_j^T x_j - 2x_i^T x_j = b_{ii} + b_{jj} - 2b_{ij}$.

Вводится дополнительное условие на искомую конфигурацию:

$\overline{x} = \frac{1}{N}\sum_{i=1}^N x_i = 0$.

Матрица Грама попарных скалярных произведений в свою очередь легко выражается:

- $\overline{x} = 0 \Rightarrow \sum_{i=1}^N b_{ij} = 0$
- $\frac{1}{N}\sum\limits_{i=1}^N d_{ij}^2 = \frac{1}{N}\sum\limits_{i=1}^N b_{ii} + b_{jj}$
  $\frac{1}{N}\sum\limits_{j=1}^N d_{ij}^2 = b_{ii} + \frac{1}{N}\sum\limits_{j=1}^N b_{jj}$
  $\frac{1}{N^2}\sum\limits_{i,j=1}^N d_{ij}^2 = \frac{2}{N}\sum\limits_{i=1}^N b_{ii}$
- $b_{ij} = -\frac{1}{2}(d_{ij}^2 - d_{i\bullet}^2 - d_{\bullet j}^2 + d_{\bullet\bullet}^2)$,
  где $d_{i\bullet}^2 = \frac{1}{N}\sum\limits_{i=1}^N d_{ij}^2,\ d_{\bullet j}^2 = \frac{1}{N}\sum\limits_{j=1}^N d_{ij}^2,\ d_{\bullet\bullet}^2 = \frac{1}{N^2}\sum\limits_{i,j=1}^N d_{ij}^2$

Другими словами,

$$B = C_N A C_N, \tag{11}$$

где A получается из D поэлементным возведением в квадрат и домножением на -0.5, $C_N = E - \frac{1}{N}\mathbf{1}\mathbf{1}^T$, $\mathbf{1}\mathbf{1}^T$ — матрица из единиц.

Пусть $X \in \mathbb{R}^{N\times q}$ — матрица, в которой векторы искомой конфигурации $x_1, ..., x_N$ записаны по строкам.

Тогда по определению матрицы Грама $B = XX^T$, причем

- B — симметричная $(XX^T)^T = XX^T$
- B — неотрицательно определенная $\langle XX^T a, a\rangle = \langle X^T a, X^T a\rangle \geqslant 0$
- $rgB = rgXX^T = rgX = q$

B имеет q положительных собственных значений и $n-q$ нулевых, и может быть представлена в виде:

$$B = \Gamma\Lambda\Gamma^T, \tag{12}$$

где $\Lambda = diag(\lambda_1, ..., \lambda_q)$ и $\Gamma = (\gamma_1, ..., \gamma_q)$ — матрица собственных векторов, отвечающих собственным значениям $\lambda_1, ..., \lambda_q$.

Наконец, отсюда находим искомую конфигурацию

$$X = \Gamma\Lambda^{\frac{1}{2}} \tag{13}$$

Таким образом, решение cMDS заключается в следующем:

(a) По матрице попарных расстояний D восстанавливаем матрицу Грама B (11).

(b) Находим спектральное разложение матрицы B (12).

(c) Восстанавливаем искомую конфигурацию $x_1, ..., x_N$ (13).

Мы довольно подробно остановились на cMDS, так как этот подход является базовым подходом задач многомерного шкалирования, предлагает эффективное аналитическое решение, не требующее численного решения сложных оптимизационных задач.

cMDS гарантирует нахождение точной конфигурации, реализующей матрицу попарных расстояний в терминах (1), однако размерность q такой конфигурации может быть большой для поставленных задач. В случае, когда нужна конфигурация фиксированной размерности p нужно оставить p наибольших собственных чисел матрицы B.

Более того, cMDS часто применяют и в более общих постановках задач. Алгоритм решения cMDS может быть выполнен даже при невыполнении требования на евклидовость исходной функции различия, в этом случае у матрицы B могут появиться отрицательные собственные значения, нужно оставить p наибольших положительных собственных чисел. В таком случае, вновь не гарантируется нахождение точной реализации (1), лишь приближенной.

cMDS не может быть применен в случае, когда множество пар M не образует множество всевозможных пар. Другими словами, в матрице попарных различий D есть пропуски. По сути допускается применение cMDS в рамках задачи

метрического многомерного шкалирования, хотя формально cMDS и не решает соответствующую оптимизационную задачу (вернее находит ее точное решение лишь при евклидовости исходной функции различия).

2. *Метрическое многомерное шкалирование (metric multidimensional scaling, mMDS):*

   В метрическом многомерном шкалировании нет столь строгих требований (как в cMDS) на заданные различия $d(x, y)$: $\{(i, j, d_{ij})\}_{(i,j)\in M}$, где $d_{ij} = d(s_i, s_j)$ — в матрице попарных различий допускаются пропуски.

   Большинство подходов метрического многомерного шкалирования сводится к оптимизации функционала (6) тем или иным методом.

   Самый классический подход, предложенный [10], подразумевает оптимизацию градиентным спуском.

   Популярным методом решения оптимизационной задачи (8) является алгоритм оптимизации SMACOF [5] (Scaling by MAjorizing a COmplicated Function). В постановке оптимизационной задачи SMACOF разрешается добавить веса в функционал *Stress*, при этом факт отсутствия пары (i, j) в множестве пар M, на которых задана величина функции различия, реализуется установкой веса таких пар в ноль $w_{ij} = 0$:

$$Stress(x_1, ..., x_N) = \sum_{i,j=1}^{N} w_{ij}(d_{ij} - ||x_i - x_j||)^2 \longrightarrow \min_{x_1,...,x_N\in\mathbb{R}^p} \tag{14}$$

   Идея этого метода состоит в нахождении вариационной верхней оценки на функционал *Stress* ($g(x, \xi)$ — вариационная верхняя оценка функции $f(x)$, если $f(x) \leqslant g(x, \xi)\, \forall x \in X, \forall \xi \in Z$ и $\forall x \in X\, \exists \xi \in Z$, т.ч. $f(x) = g(x, \xi)$). Вместо оптимизации функционала *Stress* (14) предлагается итерационно оптимизировать его вариационную верхнюю оценку.

3. *Неметрическое многомерное шкалирование (non-metric multidimensional scaling, nMDS)*

   Классический подход, предложенный [10], работает с входными различиями δ_{ij}. Вводится функционал стресса:

$$S(x_1, ..., x_N) = \sqrt{\frac{\sum(f(\delta_{ij}) - \rho(x_i, x_j))^2}{\sum \rho(x_i, x_j)^2}},\ x_1, ..., x_N \in \mathbb{R}^p \tag{15}$$

В такой постановке задачи требуется найти конфигурацию, минимизирующую функционал S, а также монотонную функцию f такую, что $f(\delta_{ij}) \leqslant f(\delta_{kl})$ в случае $\delta_{ij} < \delta_{kl}$.

В общем случае такая задача не имеет точного решения, вновь необходимо решать итерационно. Дополнительная трудность возникает с функцией f, которая, вообще говоря, не определена заранее и требует нахождения. В классике проблему отыскания функции f предлагается решить методами *изотонической регрессии*: найти $\hat{d}_{ij}$ такие, что в случае $\delta_{i_1,j_1} \leqslant \delta_{i_2,j_2} \leqslant ... \leqslant \delta_{i_m,j_m}$ выполняется $\hat{d}_{i_1,j_1} \leqslant \hat{d}_{i_2,j_2} \leqslant ... \leqslant \hat{d}_{i_m,j_m}$. После этого вновь конструируется уже привычный функционал стресса:

$$S(x_1, ..., x_N) = \sqrt{\frac{\sum(\hat{d}_{i,j} - \rho(x_i, x_j))^2}{\sum \rho(x_i, x_j)^2}},\ x_1, ..., x_N \in \mathbb{R}^p \tag{16}$$

Остановимся подробнее на том, зачем в (16) знаменатель. При масштабировании конфигурации в k раз ($x_i \to kx_i$) различия, в том числе и $\hat{d}_{i,j}$, увеличатся в k раз, а весь числитель под корнем в k^2 раз. Очевидно, что при этом порядок на расстояниях построенной конфигурации никак не изменился, поэтому, чтобы это скомпенсировать, добавляется знаменатель.

В некоторых реализациях (например, manifold.MDS в пакете sklearn) в функционале оставляют лишь квадраты ошибок

$$S(x_1, ..., x_N) = \sum(\hat{d}_{i,j} - ||x_i - x_j||)^2,\ x_1, ..., x_N \in \mathbb{R}^p \tag{17}$$

После чего оптимизируют его с помощью минимизации вариационной нижней оценки алгоритмом SMACOF. Вопрос корректности такого допущения остается открытым.

Во-многом, такая постановка задачи неудачна. Суть задачи сводится к восстановлению порядка на величинах функции различия, однако, это делается через введение дополнительной неизвестной функции f.

3.1 Предлагаемый подход

Нашей целью является предложить универсальную модель, в зависимости от модификации решающую задачи метрического и неметрического многомерного шкалирования, при этом допускающую расширения на случай разнородности данных.

За основу предлагаемого подхода возьмем следующую модель машинного обучения: пусть отображение объектов в целевое пространство $l_w : \{1, ..., N\} \to \mathbb{R}^p$ осуществляется одним полносвязным слоем нейронной сети без активации с весами w. Другими словами, на вход поступает one-hot закодированный вектор (вектор с единицей в позиции, соответствующей объекту, и нулями в остальных позициях), на выходе возвращается его векторное представление размерности p. По сути, для объекта с номером i сеть будет хранить p чисел — веса, соответствующие i-му входу.

Будем использовать логику работы *сиамских сетей*. Например, для вычисления расстояния $\rho(x_i, x_j)$ мы параллельно подаем на вход объекты i и j в две сети, разделяющие общие веса (разумеется, на практике это реализуется одной сетью), получаем представления x_i, x_j, считаем на них расстояние. Представление, полученное сетью для i-го объекта, будем обозначать $l_w(i)$. Обучать сеть l_w будем обратным распространением ошибки.

Тогда для задачи *метрического многомерного шкалирования* будем оптимизировать функцию потерь stress loss:

$$L_{mMDS}(x_i, x_j) = (d_{ij} - \rho(x_i, x_j))^2 \tag{18}$$

$$Stress = \sum_{(i,j)\in M} L_{mMDS}(x_i, x_j) \longrightarrow \min_{x_1,...,x_N \in \mathbb{R}^p} \tag{19}$$

В случае задачи *неметрического многомерного шкалирования* мы отойдем от классической постановки функционала $Stress$ к предлагаемой постановке задачи (10):

$$\forall\, (i,j), (k,l) \in M \;\; d_{ij} < d_{kl} \Longleftrightarrow \rho(x_i, x_j) + m < \rho(x_k, x_l),$$

В такой постановке задачи хорошо прослеживается суть — сохранить отношение порядка на различиях. Мы предложим два подхода к решению задачи с использованием рассматриваемого полносвязного слоя l_w.

1. Рассмотрим понятие *triplet loss*. Пусть по входным данным сформирована тройка — точка интереса (anchor point), точка того же класса, что точка интереса (anchor-positive), точка другого класса (anchor-negative). Тогда triplet loss выглядит так:

$$L = max(0, d_{ap} - d_{an} + m) \tag{20}$$

Здесь d_{ap} — расстояние от представления anchor point до представления anchor-positive, d_{ap} — до представления anchor-negative соответственно, m — отступ, гарантирующий выполнение $|d_{ap} - d_{an}| \geq m$.

В нашей задаче отсутствуют метки anchor-positive и anchor-negative на объектах. Однако, мы можем использовать в качестве anchor-positive и anchor-negative для объекта с номером i такие объекты с номерами j и k соответственно, что $d_{ij} < d_{ik}$.

Таким образом, пусть пары $(i, j), (i, k) \in M$ и $d_{ij} < d_{ik}$:

$$L_{nMDS}(x_i, x_j, x_k) = max(0, \rho(x_i, x_j) - \rho(x_i, x_k) + m), \tag{21}$$

Тогда функция потерь будет выглядеть так:

$$L_{nMDS} = \sum_{(i,j),(i,k) \in M} L_{nMDS}(x_i, x_j, x_k) \tag{22}$$

2. Естественным образом появляется предложение работать с четверками объектов. Рассмотрим понятие *quadruplet loss*. Пусть пары $(i, j), (k, l) \in M$ и $d_{ij} < d_{kl}$:

$$L_{nMDS}(x_i, x_j, x_k, x_l) = max(0, \rho(x_i, x_j) - \rho(x_k, x_l) + m) \tag{23}$$

Тогда функция потерь будет выглядеть так:

$$L_{nMDS} = \sum_{(i,j),(k,l) \in M} L_{nMDS}(x_i, x_j, x_k, x_l) \tag{24}$$

3.2 Сравнительный анализ предлагаемого подхода

В классике ключевым функционалом качества построенной конфигурации методами многомерного шкалирования является нормированная величина функционала стресса (например, (7) в случае метрического многомерного шкалирования и (16) в случае неметрического).

Помимо величины функционала стресса будем использовать ранговую корреляцию Спирмена, как показатель того, насколько хорошо сохраняется порядок на различиях. Обозначать предлагаемые методы будем префиксом «fc» (от fully-connected — полносвязный).

Для сравнения подходов сгенерирован набор данных $s_1, ..., s_N \in \mathbb{R}^D$ при $D = 3$. Функция различия $d(s_i, s_j) = \rho_E(s_i, s_j)$ — евклидово расстояние, размерность искомой конфигурации $p = 2, 3$. Сравниваем величину нормированного функционала стресса и ранговую корреляцию Спирмена. Реализация на PyTorch.

Решение задачи метрического многомерного шкалирования:

Размерность конфигурации p = 3		
Метод	Stress	Корреляция
cMDS	10^{-16}	1.0
SMACOF	10^{-3}	1.0
fc-mMDS	10^{-3}	1.0

Размерность конфигурации p = 2		
Метод	Stress	Корреляция
cMDS	0.277	0.811
SMACOF	0.218	0.849
fc-mMDS	0.222	0.845

Таблица 1: Сравнительный анализ методов метрического многомерного шкалирования.

Решение задачи неметрического многомерного шкалирования:

Размерность конфигурации p = 2	
Метод	Корреляция
cMDS	0.811
sklearn nMDS	0.017
SMACOF	0.849
triplet fc-nMDS	0.848
quadruplet fc-nMDS	**0.851**

Таблица 2: Сравнительный анализ методов неметрического многомерного шкалирования.

Прокомментируем полученные результаты. В случае, когда известно, что существует точное решение (размерность $p = 3$, таблица 1), метод успешно находит точное решение. В случае размерности искомой конфигурации $p = 2$ (таблица 1) метод обходит базовый метод cMDS, и сопоставим со SMACOF.

Для задачи неметрического многомерного шкалирования метод с triplet loss показывает сопоставимое со SMACOF качество, а метод с quadruplet loss и вовсе превосходит остальные методы.

4 Задача построения векторных представлений слов

Для демонстрации универсальности предложенной модели, а также возможности работать с разнородными данными выберем конкретную прикладную область. Рассмотрим *задачу построения векторных представлений слов*.

Будем работать со словами естественного языка. Пусть задан *словарь* V — множество слов. Задача построения векторных представлений заключается в нахождении и сопоставлении конфигурации $x_1, ..., x_N \in \mathbb{R}^p$ всем словам из словаря V таким образом, что $x_1, ..., x_N$ отвечают некоторым критериям качества. Найденные векторы x_i конфигурации будем называть *векторными представлениями слов*.

4.1 Оценивание векторных представлений слов

Следует определиться, что будем считать критериями качества найденной конфигурации. Другими словами, как оценивать построенные векторные представления. Вслед за [2] поделим методы оценивания на два класса:

1. *Внешнее оценивание (extrinsic evaluation)*

   Общая идея методов этого класса заключается в оценивании качества векторных представлений при решении «внешних» задач с помощью других методов машинного обучения и обработки естественного языка.

   Например, при решении задачи анализа тональности текста (оценки эмоциональной окраски) или задачи частеречной разметки (проставление метки части речи каждому слову текста) могут применять векторные представления слов. В таком случае, существенное улучшение решений этих задач будет говорить о качественном улучшении векторных представлений.

2. *Внутреннее оценивание (intrinsic evaluation)*

   Подход методов этого класса заключается в непосредственном оценивании качества векторных представлений вне контекста задач обработки естественного языка. Часто векторные представления оценивают, опираясь на представления людей о взаимосвязях слов.

   В настоящее время принято выделять два основных свойства векторных представлений: *свойство семантической близости* и *сохранения аналогий*.

Остановимся подробнее на каждом свойстве. Неформально, свойство семантической близости означает близость в математическом смысле у полученных представлений для близких по смыслу слов. Отдельный вопрос, что понимать под близостью слов по смыслу. Чаще всего, выделяют два понятия: близость в смысле похожести (similarity) и связанность (relatedness). Объясним разницу этих понятий на примере. Нам знакомо понятие слов синонимов. По сути, это разные словесные описания одной и той же сущности. Далее, существуют слова, описывающие очень близкие вещи или категории, например, crocodile и alligator. По сути, такие слова можно считать почти синонимами, очень близкими в смысле похожести (similarity). Продолжая эту логику, слова car и crash не будут считаться близкими, так как описывают непохожие сущности. Однако, вполне естественно заметить, что слова автомобиль и авария в определенном смысле связаны (related). Это слова одной темы, не зря связанность часто определяют как близость тем или доменов (topical similarity, domain similarity). Еще пример, для слов car и train естественно говорить о близости, так как объекты, которые обозначают эти слова во-многом похожи (функция, материал, движение, колеса, окна и т.д.). Напротив, для слов Freud и psychology естественно говорить о связанности.

Основная идея свойства сохранении аналогий (word analogy task) заключается в следующем: задаются некоторые два связанных слова a и b, и некоторое другое слово c. По полученным представлениям тем или иным образом строится новое слово d, связанное с c также, как a связано с b. Таким образом, сохранение аналогий заключается в способности представлений строить слова по аналогии. Например, если задать в качестве слова a = «man», а слова b = «woman», то для слова c = «uncle» естественно ожидать построения слова «aunt».

В этом исследовании будем работать с методами внутреннего оценивания.

Обычно выполнение свойства семантической близости проверяют с помощью заранее размеченных экспертами наборов данных — «золотых стандартов» семантической близости. Общий подход построения таких наборов — тем или иным способом оценить семантическую близость на парах слов.

По уже построенному «золотому стандарту» оценивают представления с помощью ранговой корреляции Спирмена. Методы оценивания с помощью задачи аналогии обсудим позже.

Пусть известны степени семантической близости на парах слов M: $\{(i, j, s_{ij})\}_{(i,j)\in M}$. Можно действовать двумя путями: определить функцию близости $s(x, y)$ (часто косинусная близость) на построенный векторных представлениях, либо преобразовать близости s_{ij} в различия d_{ij}. В первом случае строят вариационный ряд на значениях $s(x_i, x_j)$ и на известных семантических близостях s_{ij}. Во втором случае строят вариационный ряд на значениях $d(x_i, x_j)$ и на полученных различиях d_{ij}. После чего осуществляется переход к рангам в вариационных рядах и вычисляется коэффициент корреляции Спирмена:

$$\rho = 1 - \frac{6}{n(n-1)(n+1)} \sum_{k=1}^{|M|} (R_k - S_k)^2, \tag{25}$$

где R_k, S_k - ранги s_{ij} и $s(x_i, x_j)$ (d_{ij} и $d(x_i, x_j)$) в соответствующих вариационных рядах.

Существует еще один подход внутреннего оценивания векторных представлений — категоризация (подход упомянут в [14]). Предполагается, что задан некоторый набор данных, состоящий из M слов и K кластеров. Идея состоит в попытке восстановить некоторым алгоритмом кластеризации эти кластеры, но в пространстве векторных представлений слов. Оценкой качества служит среднее или суммарное расстояние между восстановленными кластерами и заданными.

4.2 Анализ существующих «золотых стандартов» семантической близости

Название	Язык	Количество пар	Диапазон оценки	Значения оценки
WordSim353	Английский	353	[0; 10]	Вещественные
SimLex-999	Английский	999	[0; 10]	Вещественные
The MEN Test Collection	Английский	3000	[0; 50]	Целые

Таблица 3: Общие характеристики «золотых стандартов».

Первоначально выбраны три самых популярных набора данных, связанных с оцениванием качества представлений. Общие характеристики см. в таблице 1.

Подробнее остановимся на происхождении этих наборов данных (датасетов). Мы уже обсуждали принципиальную разницу близости (similarity) и связанности

(relatedness). «Золотые стандарты» обычно получают путем усреднения оценок экспертов разметчиков. Перед началом процедуры разметки экспертам предоставляют инструкцию, в которой описан подход к оцениванию похожести или связанности слов.

- WordSim353 [8] — самый старый из представленных наборов данных, выпущен Evgeniy Gabrilovich в 2002 году. 353 пары слов в датасете были разбиты на две части: часть из 153 пар была размечена 14 экспертами, оставшаяся часть из 200 слов размечена 16 экспертами. Каждого эксперта просили оценить по шкале от 0 до 10 связанность (relatedness) слов в паре, а затем оценку усредняли между экспертами. В оригинальном датасете представлены оценки каждого конкретного эксперта, что можно использовать для получения лучшей оценки, чем обычное усреднение. Например, можно учитывать оценку каждого эксперта с весом, корректирующим смещение оценок эксперта в ту или иную сторону. Опционально, существует расширение датасета, с проставлением метки каждой паре об отношениях между словами в паре: identical tokens, synonym, antonym, hyponym, hyperonym, sibling terms, first is part of the second one, second is part of the first one, topically related.

  Значительный недостаток этого набора — использование только существительных. Кроме этого, в инструкции не объясняется разница между similarity и relatedness, поэтому есть некоторая путаница между этими понятиями, следствием чего является нечувствительность к этой разнице в оценках датасета. Существуют модификации датасета с делением пар на similar и related.

- The MEN Test Collection [3] — самый крупный из представленных набор данных с 3000 парами слов, собранный Elia Bruni в 2012 году. В этом наборе уже представлены слова трех частей речи: существительные, глаголы и прилагательные (причем внутри пары могут встречаться слова разных частей речи). Оценки парам получены другим способ, нежели в WordSim353: вместо того, чтобы просить эксперта оценить в некоторой шкале связанность в паре, эксперту предлагают выбирать между двумя парами, в какой слова более связаны. Оценка каждой паре получается в результате работы одного эксперта, здесь могут быть потенциальные проблемы, отсутствует согласование мнений нескольких экспертов. Каждому эксперту предоставляется пара слов и 50 случайно выбранных других пар, а затем от эксперта ожидается 50 сравнений, в какой паре слова более связаны. В результате, в случае наиболее связанных слов в паре, эксперт

во всех случаях выберет эту пару в сравнении. Таким образом, пара получит наибольшую оценку 50.

Ключевой недостаток этого набора данных остается прежним — близость (similarity) считается авторами частным случаем связанности (relatedness). Экспертам вновь не объясняется разница между этими понятиями, поэтому в наборе могут возникать артефакты, связанные с нечувствительностью к близости и связанности.

- SimLex-999 [9] — самый новый из представленных наборов данных, выпущен в 2014 году. Изначально, датасет задумывался с целью решить ключевую проблему WordSim и MEN — нечувствительность к разнице близости и связанности. Принцип получения оценок из мнений экспертов здесь такой же, как в WordSim353, но экспертам теперь явно в инструкции объясняется разница между similarity и relatedness, экспертов просят оценить именно similarity. Например, для similar слов 'coast' - 'shore' оценки 9.00 (SimLex-999) 9.10 (WordSim353), а для related слов 'clothes' - 'closet' оценки 1.96 (SimLex-999) 8.00 (WordSim353). В этом наборе есть слова разных частей речи: 666 пар с существительными, 222 пары с глаголами, 111 пар с прилагательными. Недостатком здесь будет отсутствие пар с разными частями речи в паре, как в MEN.

  В наборе дополнительно есть следующие теги: часть речи, рейтинги конкретности обеих слов (concreteness rating), сила ассоциации между словами (The strength of free association from word1 to word2), стандартное отклонение всех оценок экспертов на этой паре (The standard deviation of annotator scores when rating this pair) — можно использовать для оценки уверенности в оценке близости.

Резюмируем результаты анализа в таблице 2.

Название	Что оценивалось?	Число разметчиков	Распределение по частям речи
WordSim353	Relatedness	13 на 153 пары/16 на 200 пар	100% n
The MEN Test Collection	Relatedness	1 на каждую пару	81.5% n + 12.76% j + 5.73% v
SimLex-999	Similarity	500	66.6% n + 11.1% j + 22.2% v

Таблица 4: Особенности «золотых стандартов».

Выше были упомянуты проблемы, связанные с неразличимостью similarity и relatedness. Эти проблемы являются следствием недостаточно точно сформулированных инструкций экспертам по разметке, их исправить мы не сможем.

Интересует вопрос, сохраняется ли в «золотых стандартах» свойство симметричности. Вполне естественно считать, что на паре (w_1, w_2) и на паре (w_2, w_1) мера близости должна совпадать. Проверим выполнение этого свойства в выбранных наборах данных.

- В WordSim353 — обнаружено 2 пары с симметричностью слов: (money, bank) и (tiger, tiger). Пару (tiger, tiger) создатели датасета считают дефектом, его в нем быть не должно. А на паре (money, bank) симметричность не сохраняется — оценки связанности отличаются. Кроме того, был обнаружен дубликат (money, cash).
- В MEN симметричных пар слов не обнаружено, дефектов с дублированием и, одним и тем же словом, в паре не найдено.
- В SimLex-999 найдена одна пара с симметричностью слов (strange, sly), при этом оценки связанности отличаются.

Продублируем результаты в таблице 3.

Название	Симметричность	Дефекты
WordSim353	Нет; 'money-bank'	пара 'tiger-tiger'; дубликат 'money-cash'; несимметричность 'money-bank'
The MEN Test Collection	Да; отсутствуют зеркальные пары	Не обнаружено
SimLex-999	Нет; 'strange-sly'	Нет; 'strange-sly'

Таблица 5: Дефекты «золотых стандартов».

В случае WordSim353 и SimLex-999 можно удалить дефектные пары, и считать, что во всех трех наборах данных выполнено свойство симметричности (будут отсутствовать зеркальные пары).

В рамках реализации библиотеки для работы с разнородными данными описанные дефекты устранены, выбраны форматы хранения наборов на диске. Реализованы классы для представления пары слов, хранилища «золотого стандарта», «золотого стандарта», процедуры оценивания. В качестве основной структуры данных хранилища выбран словарь. Для в классе для представлений пары слов определена операция взятия хэша.

Реализован функционал нахождения распределения частей речи, нахождения словарей наборов, решения задачи аналогий, оценивания представлений (ранговая корреляция Спирмена и точность решения задач аналогий).

4.3 Задача аналогий

Остановимся подробнее на задаче аналогий. На данный момент нет общепринятой строгой формулировки аналогии. Неформально *аналогией* называют четверку слов $a : a' :: b : b'$, где слово a относится к слову a' так же, как b к b'. Например, «man» : «woman» :: «uncle» : «aunt».

Задачей аналогии будем называть следующую задачу:

По известной аналогии $a : a' :: b : b'$ строится тройка $a : a' :: b$. Требуется по векторным представлениям $x_1, ..., x_N$ слов словаря V для тройки слов $a : a' :: b$ найти слово $\beta \in V$ так, что четверка $a : a' :: b : \beta$ образует известную аналогию, то есть $\beta = b'$.

Обычно задается набор аналогий A, а качество векторных представлений оценивается как точность — доля верно решенных задач аналогии.

Определим косинус между двумя векторами (косинусная близость) как $\cos(x, y) = \frac{\langle x,y \rangle}{||x||||y||}$. Далее будем понимать под a векторное представление $x(a)$ соответствующего слова. Существует несколько подходов к решению задачи аналогии:

- Подход *3CosAdd* рассматривает аналогию как параллелограмм:

$$b' = \arg\max_{\beta \in V} \cos(\beta, b + a - a') \tag{26}$$

- Аналогично *PairDistance*:

$$b' = \arg\max_{\beta \in V} \cos(\beta - b, a' - a) \tag{27}$$

- Подход *3CosMul* [11] предлагает в предположении нормированности векторов воспользоваться линейностью скалярного произведения $\cos(\beta, b + a - a') = \cos(\beta, b) + \cos(\beta, a) - \cos(\beta, a')$, а затем перейти от сложения к умножению:

$$b' = \arg\max_{\beta \in V} \frac{\cos(\beta, b)\cos(\beta, a')}{\cos(\beta, a) + \epsilon} \tag{28}$$

- Подход *3CosAvg* [6] предлагает по всему набору аналогий A найти средний сдвиг $s = \frac{\sum_{i=0}^{m} a}{m} - \frac{\sum_{i=0}^{n} a'}{n}$, а затем добавить его к b:

$$b' = \arg\max_{\beta \in V} \cos(\beta, b + s) \tag{29}$$

Утверждается, что такой базовый подход повышает устойчивость к шуму в векторных представлениях, кроме помогает решать задачу аналогии в моделях с полисемией (учитывается несколько смыслов у слов).

- Подход *LRCos* [6] предполагает некоторую структуру в наборе аналогий A. Например, для аналогии France : Paris :: Japan : Tokyo естественно считать слова a, b принадлежащими исходному классу $source$ — страны, а слова a', b' целевому $target$ — столицы. Идея состоит в переформулировке вопроса «Что связано с Францией так же, как Токио связано с Японией?» в вопрос «Какое слово лежит в том же классе, что Токио, и ближайшее к слову Франция?». Предлагается обучить логистическую регрессию для распознавания слов целевого класса — в обучающую выборку для одного класса попадают все слова из $target$, для второго класса все слова из $source$ и прочие случайные слова. Окончательно вектор b' находится следующим образом:

$$b' = \arg\max_{\beta \in V} P(\beta \in target) \cos(\beta, b) \tag{30}$$

Аналогично «золотым стандартам» семантической близости существуют общепринятые наборы аналогий A. Перечислим основные:

1. Google Analogy. Это самый классический большой набор аналогий из 19000 четверок. Типы связи в аналогиях: страны и столицы, валюты, морфология (формы слов).

2. MSR. Набор из 8000 четверок, тип связи: морфология.

3. BATS. Самый современный и совершенный набор аналогий. Структурирован в формате пар ($source$, $target$). Сбалансирован, содержит связи следующих типов: морфология, энциклопедическая семантика (страны — столицы, вещи — цвета, мужское — женское, животные — жилища и т.д.), лексикографическая семантика (антонимы, несколько классов синонимов, гиперонимы, меронимы и т.д.)

4.4 Построение векторных представлений на основе «золотых стандартов» семантической близости и аналогий

Сформулируем следующую задачу.

Дано: размеченные экспертами наборы семантической близости на парах слов M: $\{s_{ij}\}$, $(i,j) \in M$, набор аналогий A, словари наборов пересекаются.

Задача: предложить способ построения векторных представлений слов методами многомерного шкалирования.

Набор экспертных близостей $\{s_{ij}\}$ преобразуем в различия $\{d_{ij}\}$. Будем использовать предложенную нами модель, реализующую метрическое и неметрическое многомерное шкалирование с помощью полносвязной однослойной нейронной сети l_w.

Таким образом, используя функцию потерь stress loss:

$$L_{mMDS} = (d_{ij} - \rho(x_i, x_j))^2 \tag{31}$$

в случае метрического, и функцию потерь quadruplet loss:

$$L_{nMDS} = max(0, \rho(x_i, x_j) - \rho(x_k, x_l) + m), \text{ при } d_{ik} < d_{kl} \tag{32}$$

в случае неметрического многомерного шкалирования можем решать задачу нахождения векторных представлений по семантическим близостям.

Для привлечения наборов аналогий естественно усовершенствовать модель, уже работающую с четверками. Необходимо сконструировать функцию потерь на четверке векторных представлений, при этом про отвечающие им слова известно, что они образуют аналогию. Обратимся к подходу *PairDistance*: $b' = \arg\max_{\beta \in V} \cos(\beta - b, a' - a)$. Продолжая логику подхода естественно считать, что на четверке аналогии $a : a' :: b : b'$ достигается максимум расматриваемой косинусной близости: $\cos(b' - b, a' - a) \approx 1$. Отсюда получаем функцию потерь analogy loss:

$$L_{analogy} = -\cos(b' - b, a' - a) \tag{33}$$

Тогда искомый способ построения векторных построений заключается в нахождении конфигурации $x_1, ..., x_N$ путем минимизации функции потерь:

$$L = \sum_{(i,j),(k,l) \in M} L_{nMDS} + \alpha \sum_{(i,j,k,l) \in A} L_{analogy} \tag{34}$$

Здесь α — гиперпараметр, в общем случае может быть найден по кросс-валидации.

5 Эксперименты

Для построения векторных представлений слов будем использовать следующую разнородную обучающую информацию:

- семантическая близость — MEN (набор с наиболее удачным соотношением размера словаря и числа пар слов; есть пары разных частей речи),
- набор аналогий A — BATS [13] (большое разнообразие типов связи, большое пересечение словаря с MEN, работаем со словами, которые есть в MEN)

Реализация на PyTorch. Размерность векторных представлений $p = 100$. Аналогии решаем методом PairDistance.

Метод	корр. MEN	корр. WS353	корр. SimLex
fc-mMDS	0.99	0.34	0.30
fc-nMDS	0.99	0.34	0.31
fc-nMDS + analogy	0.95	0.34	0.30
GloVe (wiki2010)	0.68	0.60	0.32

Таблица 6: Построение векторных представлений предложенными методами.

Метод	корр. MEN	точн. BATS
fc-nMDS	**0.99**	4.7%
fc-nMDS + analogy	0.95	**20.4%**

Таблица 7: Изменение показателей при привлечении аналогий.

В таблице 6 приведено сравнение показателей качества построенных представлений с методом GloVe [15], который использует для построения большие корпуса текстов. Корреляция MEN — показатель качества на обучении. Результаты сопоставимы с методами построения векторных представлений по большим корпусам, предложенный метод выглядит перспективным, особенно в случае полноценной матрицы попарных различий (без пропусков).

В таблице 7 представлено сравнение метода, решающего задачу неметрического многомерного шкалирования, без и с привлечением аналогий. При этом, привлечение аналогий влечет рост точности решения задач аналогий, закономерно с небольшим снижением корреляции на обучении.

6 Заключение

Была рассмотрена архитектура для решения задачи метрического и неметрического многомерного шкалирования, предложены специализированные функции потерь quadruplet loss и analogy loss для неметрического многомерного шкалирования и для наборов аналогий с соответствующим влиянием на архитектуру. Предложен новый подход к построению векторных представлений слов на основе разнородной обучающей информации: размеченная экспертами семантическая близость и наборы аналогий. Проанализированы уже существующие наборы разнородной обучающей информации. Для работы с наборами семантической близости и наборами аналогий разработана и реализована программная библиотека, автоматизирующая оценку качества векторных представлений слов. Рассмотрена задача оценивания векторных представлений слов, проведен обзор методов ее решения. Проведены эксперименты, на основе разнородной информации построены векторные представления слов, полученные представления оценены. Сравнение с известными методами показывает перспективность предложенного подхода. Дальнейшей задачей в рамках этого исследования может быть задача автоматического построения матрицы близостей для слов без участия экспертов. Ожидается, что в такой матрице не будет пропусков, поэтому есть потенциальная возможность улучшить показатели качества предложенного подхода.

Список литературы

[1] Sameer Agarwal, Josh Wills, Lawrence Cayton, Gert Lanckriet, David Kriegman, and Serge Belongie. Generalized non-metric multidimensional scaling. *Journal of Machine Learning Research - Proceedings Track*, 2:11–18, 01 2007.

[2] Amir Bakarov. A survey of word embeddings evaluation methods. *ArXiv*, abs/1801.09536, 2018.

[3] Elia Bruni, Nam Khanh Tran, and Marco Baroni. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 11 2014.

[4] Vanniarajan Chellappan and Kamala Krithivasan. Distance realization problem in network tomography: A heuristic approach. In *2013 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pages 1–6, 2013.

[5] Jan De Leeuw. Modern multidimensional scaling: Theory and applications (second edition). *Journal of Statistical Software*, 14, 10 2005.

[6] Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuoka. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3519–3530, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[7] Dmitriy Drusvyatskiy, Nathan Krislock, Yuen-Lam Voronin, and Henry Wolkowicz. Noisy euclidean distance realization: robust facial reduction and the pareto frontier, 2015.

[8] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. *ACM Trans. Inf. Syst.*, 20:116–131, 01 2002.

[9] Felix Hill, Roi Reichart, and A. Korhonen. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41:665–695, 2015.

[10] J. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–27, 1964.

[11] Omer Levy and Yoav Goldberg. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 171–180, Ann Arbor, Michigan, June 2014. Association for Computational Linguistics.

[12] Ulrike von Luxburg. *Statistical Learning with Similarity and Dissimilarity Functions*. Doctoral thesis, Technische Universität Berlin, Fakultät IV - Elektrotechnik und Informatik, Berlin, 2004.

[13] Anna Rogers, Aleksandr Drozd, and Satoshi Matsuoka. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. pages 8–15, 01 2016.

[14] François Torregrossa, Vincent Claveau, Nihel Kooli, Guillaume Gravier, and Robin Allesiardo. On the correlation of word embedding evaluation metrics. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4789–4797, Marseille, France, May 2020. European Language Resources Association.

[15] Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang, and C.-C. Jay Kuo. Evaluating word embedding models: methods and experimental results. *APSIPA Transactions on Signal and Information Processing*, 8, 2019.