Aduenko A.A., Strijov V.V. Optimal text placement for titles of documents in collection // Software Engineering, 2013, 3 : 21-25.

УДК 519.256

А. А. Адуенко, студент, Московский физико-технический институт, E-mail: aduenko1@gmail.ru

В. В. Стрижов , канд.т физ.-мат. наук, науч. сотр., Вычислительный центр им. А.А. Дородницына РАН, г. Москва, E-mail: strijov@ccas.ru

Алгоритм оптимального расположения названий коллекции документов[1]

В работе исследуется метод визуализации результатов тематической кластеризации коллекции документов. Матрица парных расстояний между документами оптимальным способом спроецирована на плоскость. Требуется расположить названия документов оптимальным образом. Предложена такая функция потерь, которая позволяет расположить название тем на плоскости с минимальным перекрытием. Для ее минимизации использовался алгоритмBFGS . Алгоритм проиллюстрирован примером визуализации тезисов конференции.

Ключевые слова: визуализация, тематическая классификация, коллекция документов, функция потерь, алгоритм BFGS.

Введение

Эффективным способом анализа коллекции документов является визуализация матрицы парных расстояний между документами. Каждый документ представлен точкой с подписью. Требуется спроецировать множество точек на плоскость с минимальной, в терминах некоторого функционала, потерей информации [1] и сделать подписи к спроецированным точкам.

Существуют прикладные программы, которые применяются при решении

[1] Работа выполнена при поддержке Министерства образования и науки РФ в рамках Государственного контракта 07.524.11.4002.

подобных задач. Например, для визуализации графов используется программа Graphviz (graphviz.org). Однако, несмотря на свою распространненость, эта программа визуализирует только граф связей между вершинами. Иначе говоря, на вход программы подаются только ребра между вершинами графа, а фиксировать координаты части вершин графа нельзя. По этой причине для решения рассматриваемой задачи подобные программы неприменимы.

Для визуализации коллекции текстовых документов вводится функция расстояния [2] на множестве документов. Рассмотрим словарь — множество слов, каждое из которых хотя бы раз встретилось в одном из документов коллекции. В данной работе под документом будем понимать неупорядоченное множество слов из словаря. Слова в документе могут повторяться: документ представлен в виде «мешка слов» [3]. Каждому документу поставим в соответствие вектор, содержащий информацию о словарном составе документа. Размерность этого вектора равна количеству слов в словаре. Расстояние между документами есть расстояние между векторами, соответствующими этим документам [4, 5].

Для проецирования векторов, описывающих документы коллекции, в двумерное пространство используется метод главных компонент [1]. Для определения оптимального положения подписей вводится функция потерь. В данной работе ставится задача оптимизации предложенной функции потерь. Она выполняется с помощью метода *BFGS* (англ. Broyden–Fletcher–Goldfarb–Shanno method) [7, 8]. В качестве начального приближения используется случайное приближение вблизи исходных точек. Результаты оптимизации функций сравниваются.

Предложенный алгоритм проиллюстрирован вычислительным экспериментом на данных конференции «European Conference on Operational Research, EURO-2012». В качестве коллекции документов было взято 48 тезисов докладов конференции из раздела «DEA and performance measurement».

1 Постановка задачи

Пусть $W = \{w_1, \ldots, w_n\}$ — заданное множество слов, словарь. Документом назовем неупорядоченное множество слов из W: $\{w_j\}$, $w_j \in W$ — слово в документе. Представим документ с номером i в виде вектора

$$\mathbf{x}_i = [c(i, w_1), \ldots, c(i, w_j), \ldots, c(i, w_n)]^{\mathrm{T}},$$

где $c(i, w_j)$ — число вхождений слова w_j в i-й документ. Опишем множество документов следующим образом. Задана выборка — множество n векторов признаков документов $\mathbf{x} \in \Re^n$

$$\{\mathbf{x}_i\}, i \in \{1, \ldots, m\},$$

где m – число документов в коллекции, а $\Re$ – множество неотрицательных действительных чисел.

Задана функция расстояния между векторами признаков документов –

евклидова метрика

$$\rho(\mathbf{x}, \mathbf{x}') = \sqrt{\sum_{j=1}^{n} |x_j - x_j'|^2},$$

где x_j – компонента вектора $\mathbf{x}$. Построим матрицу попарных расстояний между всеми парами векторов $\mathbf{x}_i, \mathbf{x}_k, i, k \in \{1, \ldots m\}$. Для визуализации взаимного расположения документов, описанного матрицей попарных расстояний, предлагается найти ее оптимальную проекцию на плоскость.

Документ представлен точкой на плоскости, а заголовок документа — текстом с выноской: с линией, соединяющей точку и текст. Оптимальная проекция ищется методом главных компонент. В случае если размерность пространства проекции m не больше ранга матрицы $\mathbf{X}$, матрица проекций $\mathbf{Z}$, находится как $\mathbf{Z} = \mathbf{XU}$. Здесь $\mathbf{U}$ ортонормированная матрица, $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}_m$, где $\mathbf{I}_m$ –

единичная матрица размерности m. Матрица $\mathbf{U}$ состоит из собственных векторов матрицы $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, соответствующих m максимальным собственным числам $\lambda_1,\ldots,\lambda_m$ этой матрицы [1]

Для визуализации подписей к документам поставим в соответствие каждому вектору $\mathbf{z}$, состящему из двух компонент и являющемуся строкой матрицы проекций **Z,** вектор $\mathbf{t} \in \Re^4$ с координатами, задающими прямоугольник на плоскости $\mathbf{t} = [t_1,\ldots,t_4]$. Здесь

- t_1 — координата нижнего левого угла по оси абсцисс,
- t_2 — координата нижнего левого угла по оси ординат,
- t_3 — ширина текста,
- t_4 — высота текста.

При этом допускается отрицательное значение ширины. Оно будет указывать на то, что линия выноски соединяет точку с правым нижним углом. Если ширина текста $t_3 \geq 0$, то линия выноски соединяет точку с левым нижним углом. Значения $|t_3|$ и t_4 вычисляются исходя из длины текста и размера шрифта.

Введем функцию потерь $S(\mathbf{T}|\mathbf{Z})$, определенную на матрице $\mathbf{T} = [\mathbf{t}_1^{\mathrm{T}},\ldots,\mathbf{t}_m^{\mathrm{T}}]^{\mathrm{T}}$, при заданной матрице $\mathbf{Z}$. Оптимальная матрица $\hat{\mathbf{T}}$ будет выбираться из условия минимума $S(\mathbf{T}|\mathbf{Z})$:

$$\hat{\mathbf{T}} = \arg \min_{\mathbf{T} \in \Re_+^{m \times 4}} S(\mathbf{T}|\mathbf{Z}).$$

2 Построение функции потерь

Для построения функции потерь S рассмотрим объекты трех типов: прямоугольник, соответствующий визуализируемому тексту; отрезок, соединяющий точку и текст; точку, обозначающую документ. Функция потерь

$S(\mathbf{T}\,|\,\mathbf{Z})$, зависящая от координат прямоугольников $\mathbf{T}$ при заданных координатах точек $\mathbf{Z}$, имеет вид

$$S(\mathbf{T}\,|\,\mathbf{Z}) = \frac{1}{m(m-1)}\sum_{i\neq k}\sum\left[w_\alpha\alpha(\mathbf{z}_i,\mathbf{t}_k,\mathbf{z}_k) + w_\delta\delta(\mathbf{z}_i,\mathbf{t}_i,\mathbf{t}_k)\right] +$$

$$+\frac{2}{m(m-1)}\sum_{i>k}\sum\left[w_\beta\beta(\mathbf{z}_i,\mathbf{t}_i,\mathbf{z}_k,\mathbf{t}_k) + w_\eta\eta(\mathbf{t}_i,\mathbf{t}_k)\right] +$$

$$+\frac{1}{m^2}\sum_{i,k}\sum\left[w_\gamma\gamma(\mathbf{z}_i,\mathbf{t}_k)\right] + \frac{1}{m}\sum_i\left[w_\rho\rho(\mathbf{t}_i,\mathbf{z}_i)\right], \text{ где } i,k\in\{1,\ldots m\}. \qquad (1)$$

В (1) множители перед суммами введены для нормировки соответствующих штрафов. Веса штрафов $w_\alpha, w_\delta, w_\beta, w_\gamma, w_\eta, w_\rho$ введены для управления вкладом каждого штрафа в функции потерь S. Опишем каждое слагаемое в функции (1) в отдельности. Слагаемое ρ в (1) штрафует за удаление надписи на большое расстояние от соответствующей точки.

$$\rho(\mathbf{t}_i,\mathbf{z}_k) = \left\|\mathbf{z}_k - \tilde{\mathbf{t}}_i\right\|_2, \qquad (2)$$

где $\tilde{\mathbf{t}}_i$ — вектор, содержащий координаты точки прямоугольника $\mathbf{t}_i$, в которой кончается линия выноски, $\tilde{\mathbf{t}} = [t_1, t_2]$. Это, в соответствии с ГОСТ 2.316-68 [10], может быть нижний левый или нижний правый угол прямоугольника.

Остальные пять функций $\alpha,\beta,\gamma,\delta,\eta$ отражают штрафы за наложение одного объекта на другой. В табл. 1 указано, какое слагаемое, за что отвечает.

Таблица 1: Штрафы за наложение объектов.

Объекты, на которые происходит наложение	Объекты, которые накладываются		
	Точка	Линия	Прямоугольник
Точка	–	–	–

Линия	α	β	–
Прямоугольник	γ	δ	η

По строкам таблицы 1 указаны объекты, на которые происходит наложение. По столбцам таблицы 1 указаны объекты, которые накладываются. Прочерк в ячейке таблицы означает, что данное наложение не штрафуется функцией потерь (1). Определим оставшиеся пять слагаемых в функции потерь $S(\mathbf{T}|\mathbf{Z})$.

Для этого определим расстояние l от некоторой точки $\mathbf{z}_i$ до выноски $(\mathbf{z}_k, \tilde{\mathbf{t}}_k)$, как евклидово расстояние от точки $\mathbf{z}_i$ до ее проекции $\mathbf{p}_i$ на этот отрезок

$$l = \|\mathbf{x}_i - \mathbf{p}_i\|_2.$$

Запишем $\mathbf{p}_i$ в виде

$$\mathbf{p}_i = \zeta \mathbf{x}_k + (1-\zeta)\tilde{\mathbf{t}}_k. \tag{3}$$

Из свойств проекции получаем, что

$$\zeta = \frac{(\mathbf{z}_i - \tilde{\mathbf{t}}_k)^{\mathrm{T}}(\mathbf{z}_k - \tilde{\mathbf{t}}_k)}{\|\mathbf{z}_k - \tilde{\mathbf{t}}_k\|^2}. \tag{4}$$

Значение вектора $\mathbf{p}_i$ находится по известной ζ с помощью (3).

Слагаемое-штраф $\alpha(\mathbf{z}_i, \mathbf{z}_k, \mathbf{t}_k)$ зададим в виде

$$\alpha(\mathbf{z}_i, \mathbf{z}_k, \mathbf{t}_k) = g \max\left(\frac{h-l}{h}, 0\right),$$

где h – ширина прямоугольника, и множитель

$$g = \begin{cases} 1, \text{если } \zeta \in [0,1], \\ 0, \text{в противном случае,} \end{cases} \tag{5}$$

где ζ определено в (3) и вычисляется по формуле (4). Однако определенная формулой (5) g, а значит и α, не является непрерывной по ζ, что затрудняет птимизацию α По этой причине в качестве g рассмотрим

$$g = \begin{cases} \zeta(1-\zeta), \zeta \in [0,1], \\ 0, \zeta \notin [0,1]. \end{cases}$$

Определим штраф $\beta(\mathbf{z}_i, \mathbf{t}_i, \mathbf{z}_k, \mathbf{t}_k)$ за пересечение отрезков выносок $(\mathbf{z}_i, \tilde{\mathbf{t}}_i)$ и $(\mathbf{z}_k, \tilde{\mathbf{t}}_k)$. Если отрезки не пересекаются, полагаем $\beta = 0$, иначе, находим точку $\mathbf{z}$ пересечения отрезков. Записываем ее в виде

$$\mathbf{z} = \zeta_1 \mathbf{z}_i + (1-\zeta_1)\tilde{\mathbf{t}}_i = \zeta_2 \mathbf{z}_k + (1-\zeta_2)\tilde{\mathbf{t}}_k.$$

Заметим, что когда отрезки $(\mathbf{z}_i, \tilde{\mathbf{t}}_i)$ и $(\mathbf{z}_k, \tilde{\mathbf{t}}_k)$ пересекаются, то $\mathbf{z}$ находится внутри каждого из них и $\zeta_1, \zeta_2 \in [0,1]$. Ситуация $\zeta_1 = 0;1$ или $\zeta_2 = 0;1$ соответствует касанию отрезков. Укажем, как найти точку пересечения отрезков. Для этого найдем точку пересечения прямых, содержащих эти отрезки, если таковая существует. Заметим, что в данной задаче исходные точки разные, потому разные отрезки не могут полностью совпадать. Запишем условие пересечения прямых

$$\mathbf{z}_i + \zeta_1(\tilde{\mathbf{t}}_i - \mathbf{z}_i) = \mathbf{z}_k + \zeta_2(\tilde{\mathbf{t}}_k - \mathbf{z}_k)$$

для некоторых $\zeta_1, \zeta_2 \in \Re$. Перепишем это условие в виде

$$\zeta_1(\tilde{\mathbf{t}}_i - \mathbf{z}_i) - \zeta_2(\tilde{\mathbf{t}}_k - \mathbf{z}_k) = \mathbf{z}_k - \mathbf{z}_i.$$

Последнее выражение есть система линейных уравнений на ζ_1, ζ_2

$$\mathbf{A}\boldsymbol{\zeta} = \mathbf{b},$$

где $A = \left(\tilde{\mathbf{t}}_i - \mathbf{z}_i, -\tilde{\mathbf{t}}_k + \mathbf{z}_k\right)$, $\mathbf{b} = \mathbf{z}_k - \mathbf{z}_i$. Если матрица $\mathbf{A}$ вырождена, то есть $\det(\mathbf{A}) = 0$, то отрезки параллельны. Иначе можно однозначно найти $\boldsymbol{\zeta} = \mathbf{A}^{-1}\mathbf{b}$. В этом случае прямые соответствующие отрезкам пересекаются. Тогда и только тогда, когда пересекаются сами отрезки, выполнено $\zeta_1, \zeta_2 \in [0,1]$. Определим функцию $\varphi(\xi), \xi \in \Re$

$$\varphi(\xi) = \begin{cases} \xi, \text{если } \xi > 0 \\ 0, \text{если } \xi \le 0. \end{cases}$$

С помощью φ штраф за пересечение отрезков $\beta(\mathbf{z}_i, \mathbf{t}_i, \mathbf{z}_k, \mathbf{t}_k)$ можно определить как

$$\beta(\mathbf{z}_i, \mathbf{t}_i, \mathbf{z}_k, \mathbf{t}_k) = \varphi(\zeta_1(1-\zeta_1))\varphi(\zeta_2(1-\zeta_2)).$$

Таким образом построенный штраф $\beta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{x}_k, \mathbf{t}_k)$ будет обладать

непрерывностью по $\mathbf{t}_i, \mathbf{t}_k$.

Определим штраф $\gamma(\mathbf{z}_i, \mathbf{t}_k)$ за попадание точки $\mathbf{z}_i$ внутрь прямоугольника $\mathbf{t}_k$. Обозначая $d_k = t_{k3}$ ширину прямоугольника, а $h = t_{k4}$ высоту, определим ξ_1, ξ_2 следующим образом

$$\xi_1 = \frac{|z_{i1} - \hat{t}_{k1}|}{\frac{d_k}{2}}, \xi_2 = \frac{|z_{i2} - \hat{t}_{k2}|}{\frac{h}{2}},$$

где $\hat{t}_{k1}$ и $\hat{t}_{k2}$ – координаты центров прямоугольников по осям абсцисс и ординат соответственно:

$$\hat{t}_{k1} = t_{k1} + \frac{1}{2} t_{k3}, \hat{t}_{k2} = t_{k2} + \frac{1}{2} t_{k4}.$$

В качестве штрафа $\gamma(\mathbf{z}_i, \mathbf{t}_k)$ возьмем

$$\gamma(\mathbf{z}_i, \mathbf{t}_k) = \varphi(1 - \xi_1)\varphi(1 - \xi_2).$$

Этот штраф тоже непрерывен в силу непрерывности φ и непрерывной зависимости ξ_1 и ξ_2 от $\mathbf{x}_i$ и $\mathbf{t}_k$.

Определим штраф δ за пересечение отрезка выноски и прямоугольника. Если отрезок $(\mathbf{z}_i, \tilde{\mathbf{t}}_i)$ и прямоугольник $\mathbf{t}_k$ не пересекаются, то $\delta(\mathbf{z}_i, \mathbf{t}_i, \mathbf{t}_k) = 0$. Если пересечение имеет место, рассмотрим две фигуры G_0 и G_1 на которые разбивается прямоугольник прямой p, содержащей отрезок пересечения. Вычислим площади S_0 и S_1 каждой из них и поделим меньшую S_{min} на большую S_{max}, получим $f = \frac{S_{min}}{S_{max}}$. Подсчитаем также общую длину l отрезка, находящуюся внутри прямоугольника, и полную длину L отрезка, соединяющего точки пересечения прямой p с границей прямоугольника. Определим штраф $\delta(\mathbf{x}_i, \mathbf{t}_k)$ следующим образом

$$\delta(\mathbf{z}_i, \mathbf{t}_k) = \frac{l}{L} f.$$

Определим штраф $\eta(\mathbf{t}_i, \mathbf{t}_k)$ за пересечение прямоугольников. Вычислим сначала площадь пересечения прямоугольников $\mathbf{t}_i, \mathbf{t}_k$. Заметим, что при перечении прямоугольников, стороны которых параллельны осям координат, получается также прямоугольник, стороны которого параллельны осям координат. Осталось только определить координаты его левого нижнего и правого верхнего угла. Первая координата левого нижнего угла есть максимум из соответствующих координат пересекающихся прямоугольников $\mathbf{t}_i, \mathbf{t}_k$. Вторая компонента — также максимум из соответствующих координат $\mathbf{t}_i, \mathbf{t}_k$. Координаты правого верхнего угла равны минимуму соответствующих координат правых верхних углов пересекающихся прямоугольников $\mathbf{t}_i, \mathbf{t}_k$. Тогда, добавляя условие существования прямоугольника в площадь пересечения C, получим, что

$$C = \varphi(\min(t_{i3} + t_{i1}, t_{k3} + t_{k1}) - \max(t_{i1}, t_{k1}))\varphi(\min(t_{i4} + t_{i2}, t_{k4} + t_{k2}) - \max(t_{i2}, t_{k2})). \quad (6)$$

Нормируя полученную площадь пересечения на площадь меньшего из прямоугольников, получим функцию штрафа $\eta(\mathbf{t}_i, \mathbf{t}_k)$:

$$\eta(\mathbf{t}_i, \mathbf{t}_k) = \frac{C}{\min(S_1, S_2)},$$

где C определяется формулой (6). Построенный штраф также будет непрерывным по $\mathbf{t}_i, \mathbf{t}_k$.

3 Вычислительный эксперимент

В вычислительном эксперименте предложенный алгоритм тестировался на данных 48 тезисов докладов конференции «European Conference on Operational Research, EURO-2012» из раздела «DEA and performance measurement». Минимизация функции потерь (1) производилась с помощью алгоритма *BFGS*. Для уменьшения числа шагов минимизации после каждого её шага применялись разные эвристики. (2) Приведем далее некоторые из них.

1. Стягивание отрезка выноски. Пытаемся сдвинуть выноску вдоль линии выноски в направлении к исходной точке с некоторым шагом. Стягивание происходит, если значение минимизируемой функции (1) уменьшается по сравнению с исходным.

2. Отражение. Отражаем выноску относительно вертикальной оси, проходящей через исходную точку и меняем угол прикрепления выноски. Отражение происходит, если значение минимизируемой функции (1) уменьшается по сравнению с исходным.

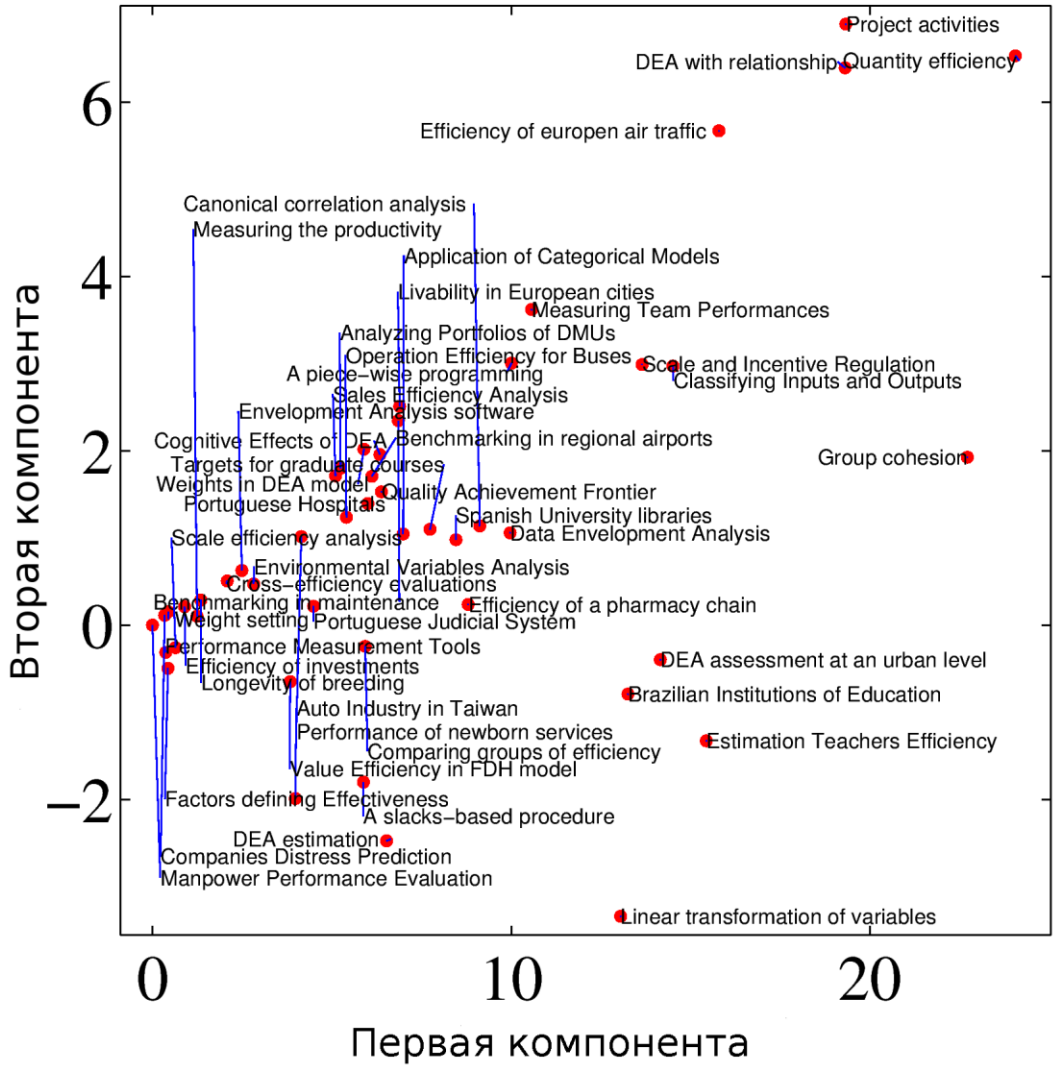

Рис. 1: Результат оптимизации функции потерь при $w_\alpha = w_\beta = w_\delta = 0$,

$w_\rho = 10, w_\eta = 10000, w_W = 10, w_H = 10, w_\gamma = 1000$.

Для того, чтобы область, где расположены точки и выноски, была как можно ближе к минимально возможной по ширине и высоте, добавим в функцию потерь S (1) штрафы за ширину W и высоту H окна, занимаемого точками. Веса, соответствующие этим штрафам обозначим w_W и w_H.

Результаты работы алгоритма при заданных значениях весов штрафов, введенных в (1) приведены на рис. 1. Площадь пересечения надписей не превосходит 0.1% от максимальной, а ширина и высота совпадают с шириной и высотой, которые определяются по исходным точкам. Это означает, что ширина и высота минимальны.

Заключение

В данной работе решена задача визуализации коллекции документов в виде точек на плоскости с подписями — названиями документов. Для проектирования векторов, соответствующих документам из коллеции, в плоскость использовались метод главных компонент. Предложена функция потерь для оптимизации расположения подписей. Для нахождения оптимального расположения используется алгоритм $BFGS$. В вычислительном эксперименте предложенный алгоритм позволил площадь пересечений выносок свести к нулю, а также оставить размер области визуализации равным минимально возможному. В этой работе описываются алгоритмы, применимые для плоского случая, однако их нетрудно обобщить на случай более высокой размерности.

Список литературы

[1] *Jolliffe I. T.* Principle Component Analysis. // New York: Springer, 2002.

[2] *Blei D. M., Ng A. Y., Jordan M. I.* Latent dirichlet allocation. // Journal of Machine Learning Research, 2003. Vol. 3. Pp. 993-1022.

[3] *Hofmann T.* Probabilistic latent semantic indexing. // Proceedings of the 22nd annual interanational ACM SIGIR conference on research and development in information retrieval. New York: ACM, 1999. Pp. 50–57.

[4] *Feldman R., Dagan I., Hirsh H.* Mining text using keywords distributions. // Journal of Intelligent Information Systems, 1998. Vol. 10(3). Pp. 281-300.

[5] *Bigi B.* Using Kullback-Leibler distance for text categorization. // Advances in information retrieval, 25th Conference on IR research. Berlin: Springer, 2003. Pp. 305–319.

[6] *Borg I., Groenen P.* Modern multidimensional scaling, theory and applications. // New York: Springer-Verlag, 1997.

[7] *Liu D. C., Nocedal J.* On the limited memory BFGS method for large scale optimiation. // Mathematical programming, 1989. Vol. 45. Pp. 503–528.

[8] *Sahari M. L., Khaldi R.* Quasi-Newton type of diagonal updating for the L-BFGS method. // Acta Mathematica Universitatis Comenianae, 2009. Vol. 78. No. 2. Pp. 173–181.

[9] *Rao R. C.* Linear statistical inference and it’s applications. // New York: Wiley, 1973.

[10] ГОСТ 2.316-68, Правила нанесения на чертежах надписей, технических требований и таблиц. // М.: ИПК Издательство стандартов, 1968.