



Факультет компьютерных наук

Прикладная математика и
информатика

Москва
2023

Автоматизация процесса подбора персонала с помощью математического моделирования

Руководитель: Наталья Титова
Выполнил: Копчев Владислав
Исследовательский проект



Проблематика



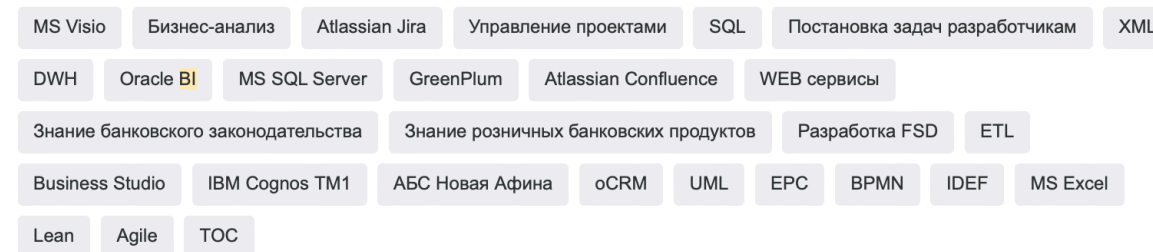


Как работают с резюме?

Два типа резюме:

- Анкета соискателя (перечисление ключевых навыков, названий университетов)
- Неструктурированный текст (описания “О себе”, опыта работы)

Ключевые навыки



Обо мне

Сильные аналитические способности, развитая интуиция.
Отличная профессиональная подготовка, опыт в своем деле 10 лет.
Представителен, вызываю доверие людей, внушаю спокойствие.
Умею слушать людей и понимать их потребности, давать им обратную связь.
Способен грамотно и понятно выражать свои мысли устно и на письме, вести протоколы и писать документацию.
Обладаю навыками решения конфликтов.
Готов разобраться в любой системе и бизнес-процессе, заняться решением задачи любой сложности.

Высшее образование (Магистр)

2007	Московский государственный технический университет им. Н.Э. Баумана, Москва Информатика и системы управления, Автоматизированные системы обработки информации и управления
2005	Казахский национальный технический университет имени К. И. Сатпаева, Алматы Информатики и информационных технологий, Информационные системы в технике и технологии

Пример резюме. Источник: https://hh.ru/resume/542ef3e70002b580e60039ed1f713943735842?query=аналитик+BI&source=search&hhtmFrom=resumes_catalog

Формальная постановка задачи

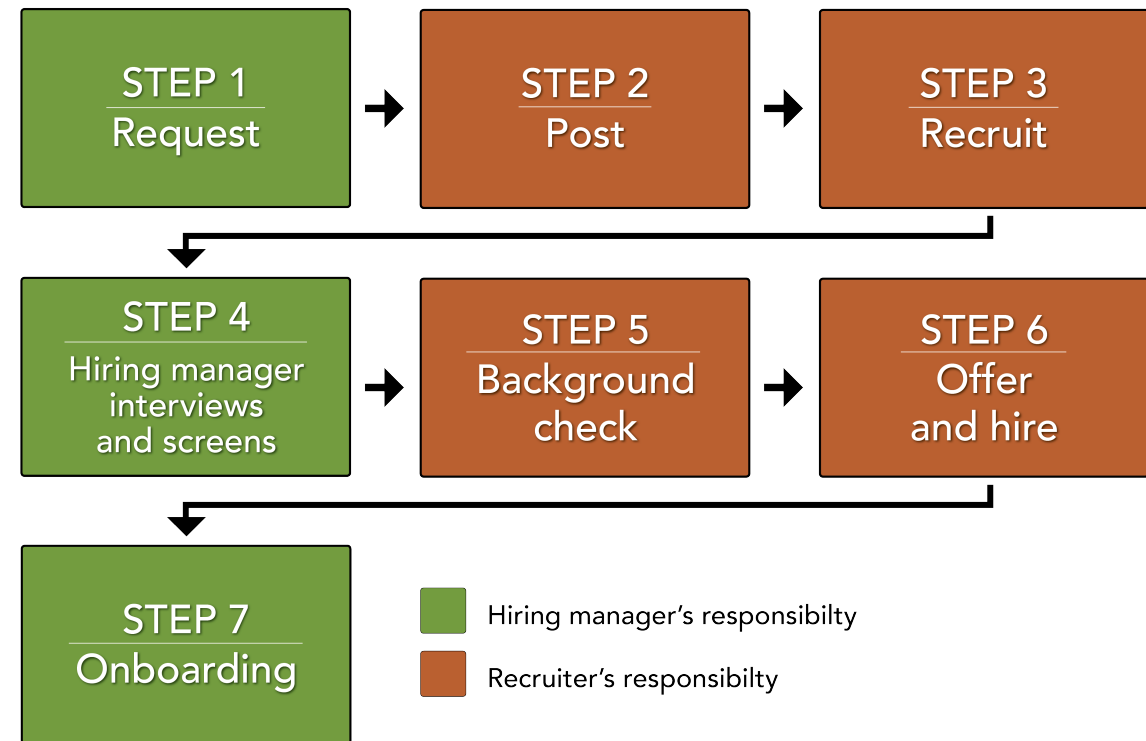
Предпосылки модели принятия решений в HR:

- Дано множество номеров кандидатов $N = \{1, \dots, n\}$
- Для каждого $i \in N$ определен вектор критериев $x_i = (x_i^1, \dots, x_i^k) \in \mathbb{R}^k$
- Для каждого $i \in N$ определены условные вероятности $p(y = 0 | x_i), p(y = 1 | x_i), p(y = 2 | x_i)$

Правило принятия решений:

- Отбираем кандидатов с наилучшими вероятностями успеха (отождествляем кандидата с вектором критериев, см. пример на картинке)

Возникает задача оценивая этих вероятностей.



Этапы принятия решений в HR. Источник: https://www.uab.edu/humanresources/home/images/LearnDev/Recruiting_OverviewHiringProcess.png



Обзор литературы

Алгоритмы классификации резюме

Pal, R., Shaikh, S., Satpute, S. и Bhagwat, S. “Resume Classification using various Machine Learning Algorithms”

SVM, Random Forest, Naive Bayes на наборе данных из 10,000 записей

Предварительная обработка NLP (tf-idf)

2022

Naidu, P. V. и др. “Resume Screening Using Machine Learning”

kNN, Logistic Regression, SVM, нейронные сети

Нейронные сети могут быть эффективно использованы

2021



Обзор литературы

Алгоритмы ранжирования резюме

Ramanath, R. и др. Towards Deep and Representation Learning for Talent Search at LinkedIn.

Нейронные сети и векторные представления
2018

Zimmermann, T. Data-driven HR - Resume Analysis Based on Natural Language Processing and Machine Learning.

Исследуются оценки: “48/100”, а не “4 года” в разделе “Опыт работы”
2016



Обзор литературы

Рекомендательные системы резюме

Roy, P. K., Chowdhary, S. S. и Bhatia, R. A Machine Learning approach for automation of Resume Recommendation system.

В работе kNN и косинусное расстояние используются для данной задачи, ранжируя резюме

2020

Barrak, A., Adams, B. и Zouaq, A. — Toward a traceable, explainable, and fairJD/ Resume recommendation system.

В работе концепции из IR, NLP объединяются для решения задачи построения системы сопоставления резюме и описания вакансий, которая дает рекомендации на основе ранжирования.

2022



Оценка вероятности успеха кандидатов

Недостатки классического решения:

1. Интерпретируемость сегментов резюме
2. Не все данные в компании размечены

HR-менеджеры смотрят не на слова («руководство», «университет», «машина», «Принстон», «обучение»).

Они смотрят на n-граммы:

- «Учился в Принстонском университете»
- «Машинное обучение с учителем».

В большинстве случаев — триграммы.

Методы, которые будем использовать:

- Частичное машинное обучение
- Кластеризация триграмм
- Суммаризация и векторные представления.



Этапы всего исследования

Рассмотрим схему этапов нашего исследования.





Постановка задачи

Цель работы: исследование методов автоматизации процесса подбора персонала

Задачи работы:

1. Построить математическую модель принятия решений в HR
2. Собрать данные (30,000 резюме) по соискателям
3. Осуществить предварительную обработку данных с сайта hh.ru (веб-скрейпинг, токенизация, суммаризация, извлечение сущностей; XPath, Selenium)
4. Разметить ~10% от собранных данных по классам 0, 1, 2 для Data Analyst и Data Scientist
5. Обучить модели машинного обучения (классификация, кластеризация; kNN, Random Forest, Logistic Regression; EM-Algorithm, KMeans, DBSCAN)
6. Осуществить валидацию моделей и интерпретацию полученных результатов с использованием топологического анализа данных и таких метрик как Silhouette Score, F_1 -мера.

Актуальность работы:

- Крайне важная в бизнесе задача, исследование методов работы с тернарными отношениями в NLP
- Возможность применения в бизнесе и разработки улучшенных методов работы с тернарными отношениями в NLP



Веб-скрейпинг резюме с hh.ru

Веб-страница — текстовый файл в формате HTML.
Данные на странице записаны в виде HTML-кода, который имеет древовидную структуру. Собирать будем с помощью Selenium, XPath.

Запросы: “аналитик BI”, “системный аналитик”, “бизнес-аналитик”, “аналитик продаж”, “финансовый аналитик”, “аналитик данных”, “data analyst”.

Столбец	Описание столбца	XPath-запрос
Название	Название резюме	<code>//div[@class="resume-search-item__header"]</code> для сбора списка всех резюме на странице с результатами поиска, <code>//div[@data-qa="resume-serp__results-search"]</code> для обращения к конкретному резюме, <code>text()</code> для обращения к тексту названия
Ссылка	Ссылка на резюме	<code>//div[@class="resume-search-item__header"]</code> для сбора списка всех резюме на странице с результатами поиска, <code>//div[@data-qa="resume-serp__results-search"]</code> для обращения к конкретному резюме, <code>href</code> для обращения к ссылке

Описание таблицы *resumes_all.csv*



Предварительная обработка данных, извлечение сущностей

Данные пока не соответствуют нашей модели. Необходимо с помощью регулярных выражений разделить данные на простые, привести к единому формату.

Будем работать с выборкой в 3000 резюме.

Признак	Проблема	Исправление
Опыт	Вместо числа лет и месяцев значение имеют вид “Опыт работы: 5 лет 6 месяцев”, “Опыт работы: 3 года”, “Work experience: 5 months” и т. д.	Убрать слова “Опыт работы”, “work experience”. Слова “лет”, “год”, “года”, “year”, “years” заменить на знак “;”. Аналогично поступить с “months”, “month”, “месяца” и т.д. Получим значения вида “x;y”. После этого мы преобразуем данные значения в дробные числа вида $x + y/12$.
Перевод уровня образования	Один и тот же уровень образования обозначается множеством различных способов. К примеру, “Bachelor”, “bachelor”, “Бакалавр” и т. д. Обозначают одно и то же.	Приведем все значения к одному из следующих нескольких: Высшее образование, Бакалавр, Доктор наук, Кандидат наук, Магистр, Неоконченное высшее образование, Образование, Среднее образование, Среднее специальное образование

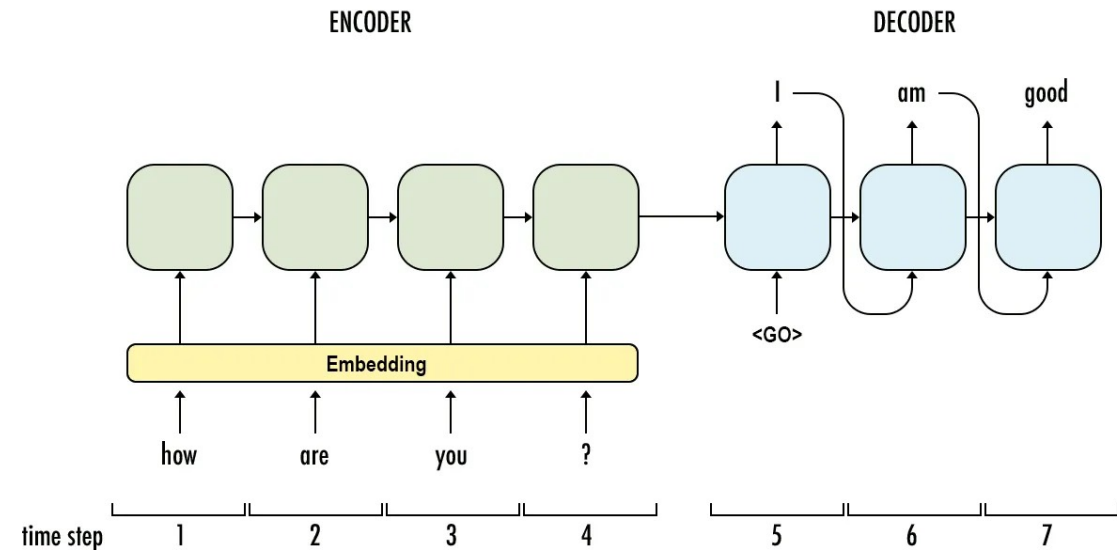
Предварительная обработка данных в таблице resumes_features. Этап 1.
Приведение к единому виду

Предварительная обработка данных, суммаризация

Теперь займемся разделом “О себе”. Необходимо выделить главную мысль в текстах с помощью метода mBART. Это seq2seq Transformer-модель, предобученная на датасете Gazeta для использования на русских текстах.

Пример:

“Люблю думать, создавать, видеть в своей работе прогресс. Не успокоюсь, пока не добьюсь поставленной цели или не разберусь в задаче. Кейс: Развитие собственного телеграмм-канала и группы ВК. Бизнес-модель: подписка. Приобретены базовые понимания: развитие проекта анализ конкурентов, поиск рекламных площадок, общение с клиентами, продажи, сведение экономики. С 5-ти лет профессионально занимался футболом. Играл в Академии Московского Спартака. Несколько лет был капитаном команды и ключевым игроком на поле. Сейчас активно играю за институт. ЯЗЫКИ ПРОГРАММИРОВАНИЯ И ЗНАНИЕ ПРОГРАММ: C, Python (базовый уровень)” → “Нахожусь в поисках работы. На данный момент работаю в телеграмм-канале ВК.”



Верхнеуровневое описание архитектуры seq2seq Tranformer. Источник <https://towardsdatascience.com/sequence-to-sequence-model-introduction-and-concepts-44d9b41cd42d>. Стоит заметить, что encoder- и decoder-компонент может быть несколько, идущих подряд.

Предварительная обработка данных, word2vec и разметка

Разметим 383 резюме — 14% всех резюме в выборке. Для аналитиков: 339 класса 0, 31 класса 1, 13 класса 2.

После этого разобьем тексты на триграммы и будем использовать такую функцию расстояния для векторов

$$\text{триграмм: } d(u, v) = 1 - \frac{(u, v)}{||u|| \cdot ||v||}.$$

Для текстов в целом – средний вектор триграмм. Заметим:

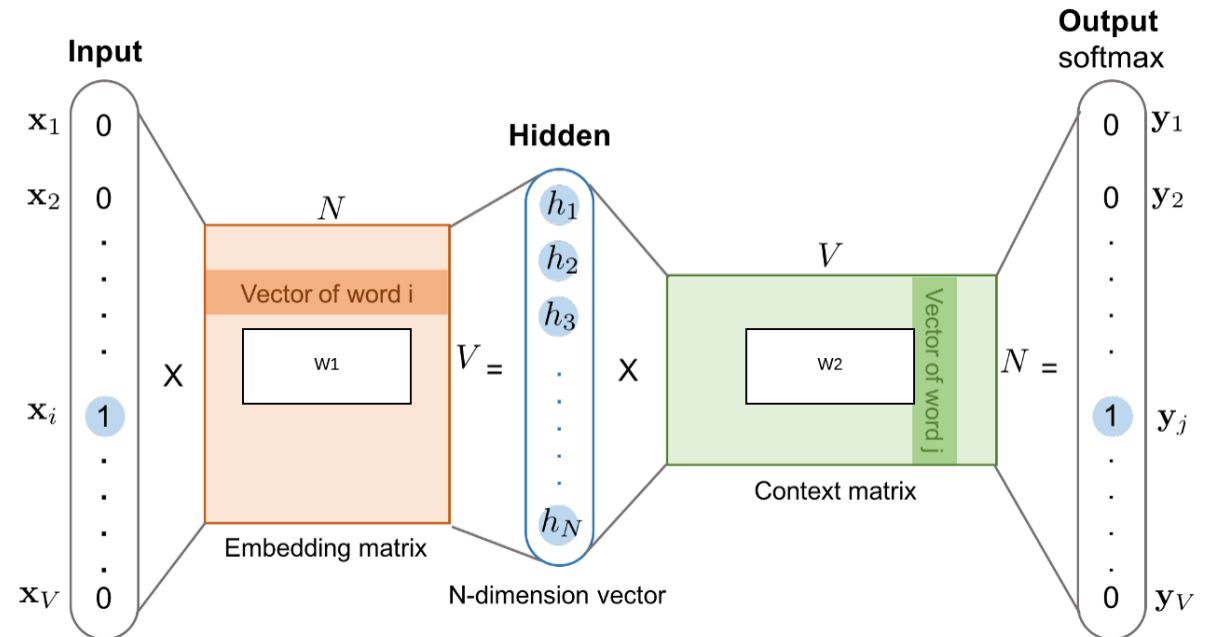
$$d\left(\frac{1}{n}(a_1 + \dots + a_n), \frac{1}{n}(b_1 + \dots + b_n)\right) = d(a_1 + \dots + a_n, b_1 + \dots + b_n) =$$

$$= d(a_{\sigma(1)} + \dots + a_{\sigma(n)}, b_{\sigma(1)} + \dots + b_{\sigma(n)}) \forall \sigma \in S_n \text{ (порядок слов не учитывается; растяжение суммы векторов)}.$$

Для близких триграмм $(a_i, b_j) \approx ||a_i|| \cdot ||b_j||$.

$$\frac{\sum_{i,j} (a_i, b_j)}{||\sum_i a_i|| \cdot ||\sum_j b_j||} \geq \sum_{i,j} \frac{(a_i, b_j)}{\sum_{i,j} ||a_i|| \cdot ||b_j||} \approx \frac{\sum_{i,j} ||a_i|| \cdot ||b_j||}{\sum_{i,j} ||a_i|| \cdot ||b_j||} = 1$$

То есть, $d(a, b) \approx 0$, если $d(a_i, b_j) \approx 0$ для многих (i, j) .



Архитектура word2vec. Источник: <https://towardsdatascience.com/word2vec-made-easy-139a31a4b8ae>



Предварительная обработка данных, итоги

Сравним наше расстояние для различных пар текстов. Видим, что наше представление о расстоянии выполняется.

Будем использовать LabelEncoder для кодирования категориальных признаков. Получили:

№	Опыт	Топовость образования	Релокация?	Город?	...	99
0	7.08	0	3	39	...	8.074022E-05
1	17.25	0	2	39	...	0.0014584048
2	18.33	0	2	91	...	-0.00043715845
3	17.41	0	0	39	...	0.00075401744
4	7.83	0	0	32	...	-0.0031653682
5	21.83	0	1	39	...	0.0015374118

Таблица с данными после предварительной обработки

Первый текст	Второй текст	Величина cosine distance между их векторными представлениями
Уверенный пользователь ПК, Ms Word, Ms Excel, 1C, Power Point, Internet Explorer, amoCRM, Bitrix24 и офисной техники; Правовых систем Гарант и Консультант +; Знание законодательства; Работа с входящей и исходящей корреспонденцией, Этика делового общения, общения с клиентами; Навыки работы по взаимодействию с государственными органами и общественными организациями; Неоднократное участие в судебных заседаниях; Анализ данных, составление договоров, запросов, отзывов, заявлений, уведомлений.	В настоящее время занимаюсь благотворительной деятельностью. В свободное время занимаюсь бегом, плаванием, рисую в стиле фантазии (карандаш), много путешествую, интересуюсь различиями культур и обычаями народов мира.	0.950825065374374
Нахожусь в отпуске по уходу за ребенком. На данный момент у меня нет официального места работы.	Нахожусь в отпуске по уходу за ребенком. На данный момент у меня нет вакантных должностей.	0.304642558097839

Таблица с примерами вычисления величины cosine distance для пар текстов из раздела "О себе"

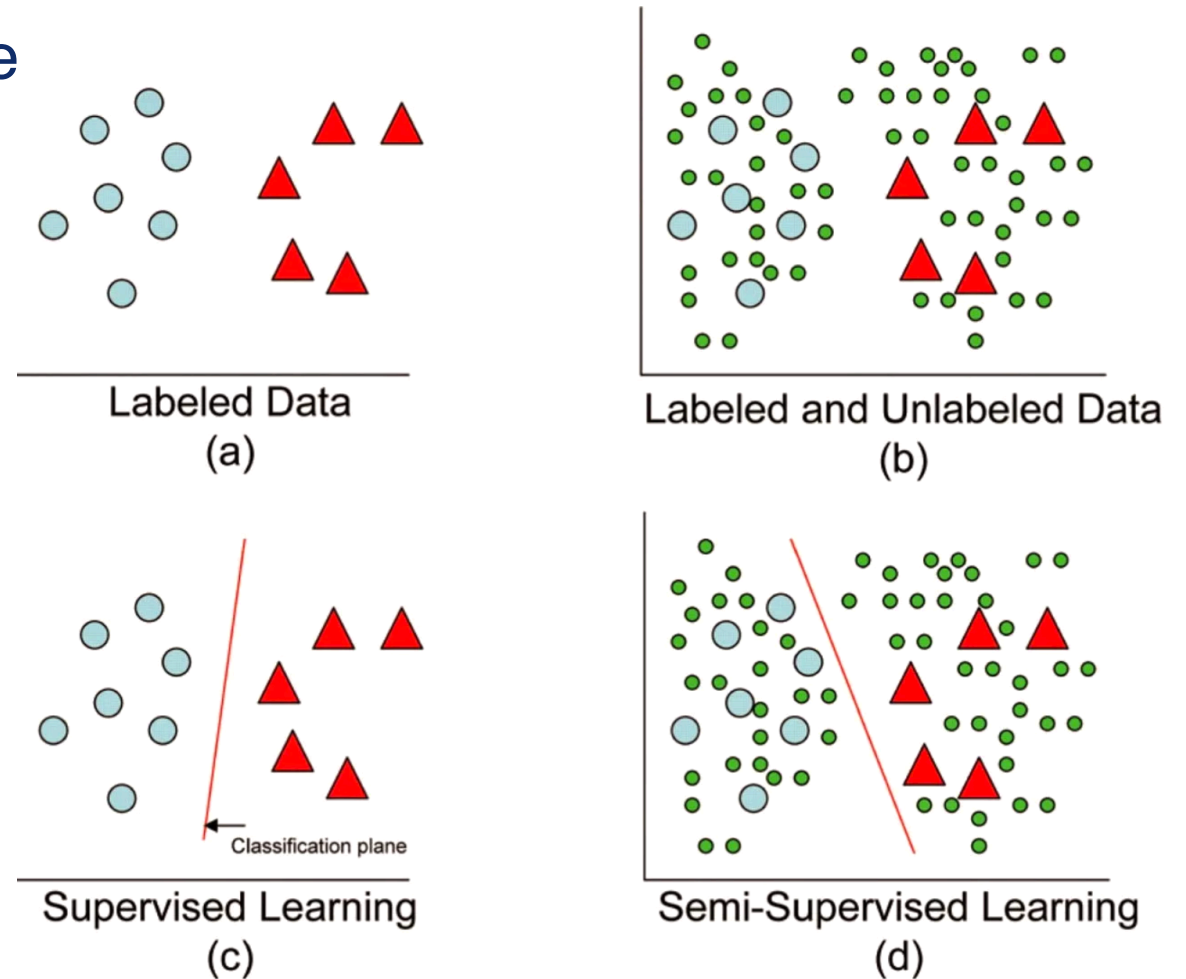
Классификация, частичное обучение

У классов 1 и 2 маленький процент среди размеченных. Поэтому модель $a(x) \equiv 0$ будет работать очень хорошо. Чтобы преодолеть проблему, **будем смотреть на F_1 -меру**, поскольку если она близка к 1, то и точность и полнота одновременно тоже.

Мы имеем обучающую выборку $X^\ell = \{(x_i, y_i)\}_{i=1}^\ell$ и незамеченную часть выборки $X^u = \{x_i\}_{i=\ell+1}^n$. При обучении модели мы хотим использовать обе части выборки. Пусть $a(x)$ — алгоритм обучения с учителем.

Метод self-training:

1. Обучим $a(x)$ на X^ℓ
2. Применим $a(x)$ к X^u
3. Добавить некоторые $(x_i, a(x_i))$ такие, что $x_i \in X^u$ в нашу выборку X^ℓ
4. Вернуться на шаг 1.





Классификация, модели

Логистическая регрессия: $p(y|x, w) = \text{Ber}(y | \text{sigm}(w^t x))$, где
$$\text{sigm}(w^t x) = \frac{1}{1 + e^{-w^t x}} = \frac{1}{1 + e^{-(w_1 x_1 + \dots + w_n x_n)}}$$

Алгоритм kNN: $p(y = c | x, k) = \frac{1}{k} \sum_{i \in N_k(x)} \mathbb{I}(y_i = c)$, где $\mathbb{I}(\cdot)$ —
индикаторная функция, $N_k(x)$ — индексы точек, входящих в
класс k наиболее близких точек.

Деревья решений. Пусть дерево имеет M листьев, R_m —
множество элементов признакового пространства, лежащих в
 m -м листе дерева (на этих элементах выдается константный
прогноз w_m). Тогда дерево решений описывается моделью

$$a(x) = \sum_{m=1}^M w_m \mathbb{I}(x \in R_m).$$
 Тогда пропорция наблюдений из

класса c в m -м листе:
$$p(y = c | x, m) = \frac{1}{|R_m|} \sum_{x_i \in R_m} \mathbb{I}(y_i = c).$$

	Logistic Regression		Decision Tree		kNN	
mean value counts	0.0	174.714	0.0	155.535	0.0	179.892
	2.0	NaN	1.0	18.035	1.0	NaN
	1.0	5.107	2.0	9.428		
min f1 micro	0.8197		0.738		0.836	
mean f1 micro	0.866		0.795		0.879	
max f1 micro	0.8961		0.836		0.907	
min f1 macro	0.307		0.296		0.303	
mean f1 macro	0.359		0.387		0.335	
max f1 macro	0.472		0.489		0.417	
min f1 weighted	0.794		0.776		0.761	
mean f1 weighted	0.836		0.807		0.835	
max f1 weighted	0.882		0.847		0.893	

Метрики качества классификации с учителем



Классификация, валидация

Основная проблема — маленькое количество размеченных единиц и двоек.

Необходимо разметить больше резюме, которые получили бы такую оценку, чтобы модели работали лучше.

Теперь необходимо произвести **кластеризацию** методами EM-алгоритма, KMeans, DBSCAN для приложений модели.

Формула: $Pr = \frac{TP}{TP + FP}$, $Re = \frac{TP}{TP + FN}$, $F_1 = 2 \cdot \frac{Pr \cdot Re}{Pr + Re}$.

$F_1^{macro} = \frac{1}{3} \cdot (F_1^1 + F_1^2 + F_1^3)$, $F_1^{weighted} = p_1 F_1^1 + p_2 F_1^2 + p_3 F_1^3$,

p_i — пропорция i -го класса, F_1^{micro} — доля верных ответов.

	LR semi-supervised	DT semi-supervised	kNN semi-supervised
mean value counts	0.0 181.714	0.0 156.428 1.0 18.392 2.0 8.178	0.0 182.357
min f1 micro	0.836	0.754	0.852
mean f1 micro	0.885	0.798	0.886
max f1 micro	0.912	0.863	0.923
min f1 macro	0.303	0.297	0.307
mean f1 macro	0.321	0.381	0.315
max f1 macro	0.429	0.474	0.371
min f1 weighted	0.766	0.755	0.784
mean f1 weighted	0.837	0.808	0.835
max f1 weighted	0.870	0.856	0.889

Метрики качества частичной классификации

Кластеризация

Перед кластеризацией применим к данным нормализацию и PCA, чтобы избежать эффекта проклятия размерности.

Пусть $x_1, \dots, x_n \in \mathbb{R}^n$ — векторы, задающие наши данные.
Задача: $\forall k \in \{0, \dots, n-1\}$ среди всех k -мерных линейных многообразий $L_k \subset \mathbb{R}^n$ найти L_k такое, что

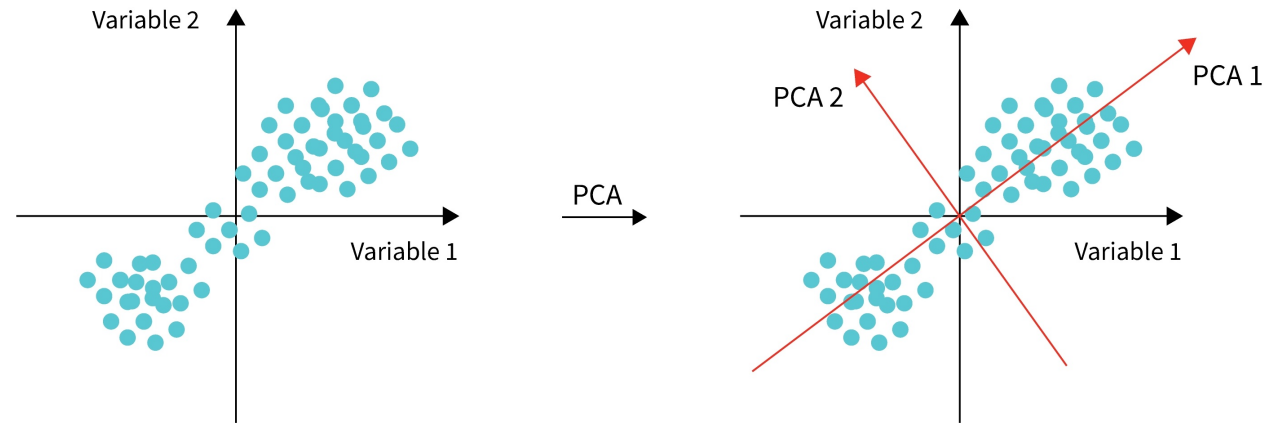
$\sum_{i=1}^m (d(x_i, L_k))^2 \rightarrow \min_{L_k}$, где $d(x, L)$ — стандартное евклидово расстояние от точки до векторного пространства.

В качестве решения задачи мы получаем ортонормированный базис a_0, \dots, a_{n-1} , которые порождают линейные многообразия $L_0 \subset \dots \subset L_{n-1}$, которые аппроксимируют наши данные.

Для векторов существует явная формула:

$$a_0 = \arg \min_{||a_0||=1} \sum_{i=1}^n ||x_i - a_0||^2, \dots,$$

$$a_k = \arg \min_{||a_k||=1} \sum_{i=1}^n ||x_i - a_k \cdot (a_k, x_i)||^2.$$



PCA при $n = k = 2$. Источник: <https://www.scaler.com/topics/nlp/what-is-pca/>

57% вариации в данных
объясняется результатом
работы PCA

Кластеризация, ЕМ-алгоритм

Для начала — модель Гауссовской смеси. Мы моделируем распределение данных как $p(x_i | \theta) = \sum_{k=1}^K \pi_k \mathcal{N}(x_i | \mu_k, \Sigma_k)$. ЕМ-

алгоритм. Пусть $\ell(\theta) = \sum_{i=1}^N \log p(x_i | \theta)$ — логарифм функции

правдоподобия. Алгоритм итеративно выполняет Е-шаг: $Q(\theta, \theta^{t-1}) := \mathbb{E}[\ell(\theta) | \theta^{t-1}]$ и М-шаг: $\theta^t := \arg \max_{\theta} Q(\theta, \theta^{t-1})$.

Пусть $a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i, j)$ — расстояние между i -й и

всеми остальными точками в C_I , $b(i) = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i, j)$ —

мин. ср. расстояние от i до остальных точек в других кластерах. Тогда $(-1, 1) \ni s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \rightarrow \max_{C_1, \dots, C_N}$.

Наилучший Silhouette Score — при количестве кластеров 6. Это 0,77 на PCA-проекции данных.

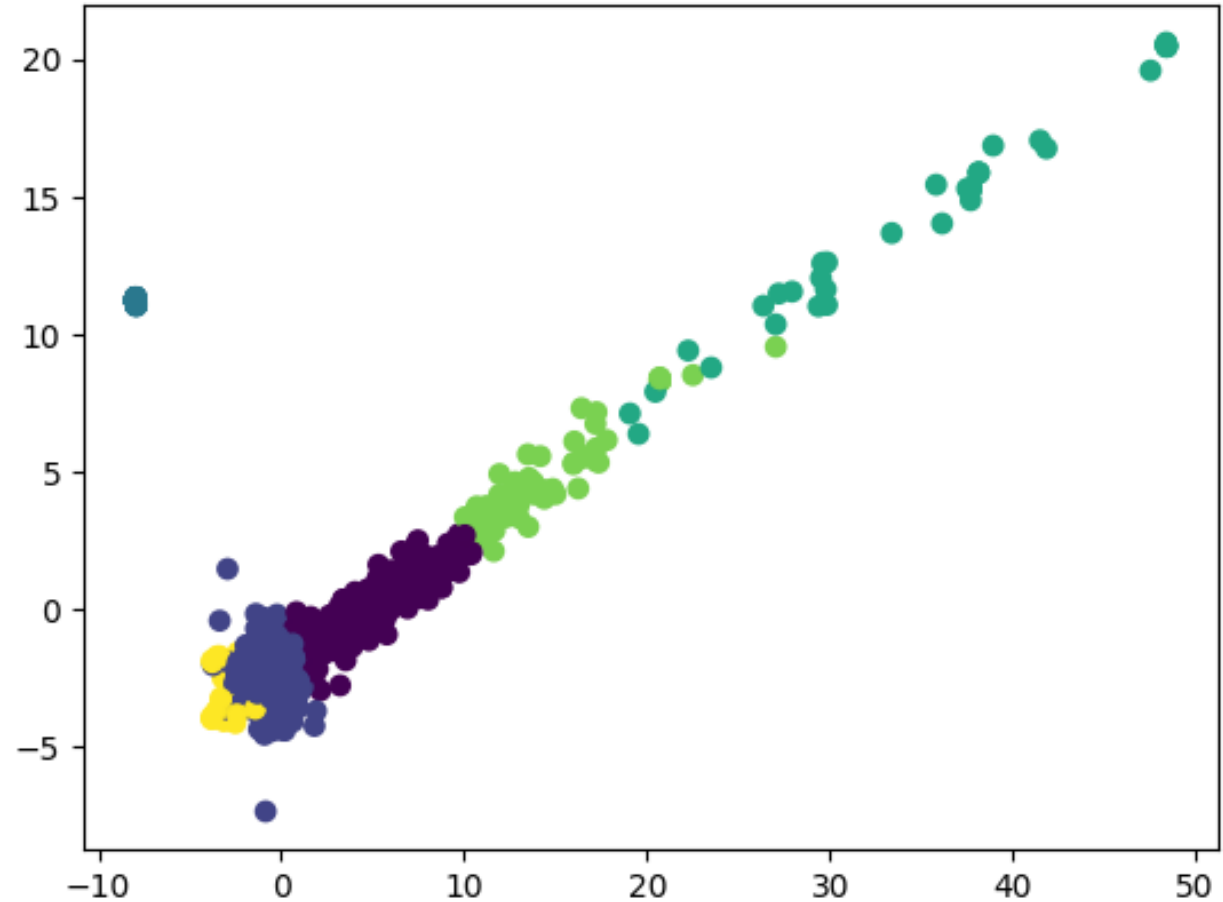


График кластеров, полученных ЕМ-алгоритмом, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров, равном 6

Кластеризация, KMeans

Алгоритм K-Means — частный случай EM-алгоритма.

Рассмотрим снова модель Гауссовской смеси, однако положим

$\pi_k = \frac{1}{K}$, $\Sigma_k = \sigma^2 I$. В результате только средние $\mu_k \in \mathbb{R}^d$ должны быть оценены.

Мы оценим их с помощью следующего алгоритма:

1. Инициализируем m_k .
2. Повторять до сходимости:
 3. 2.1. $z_i := \arg \min_k \|x_i - \mu_k\|_2^2$

4. 2.2. $\mu_k := \frac{1}{N_k} \sum_{i: z_i=k} x_i$

Посмотрим на **Silhouette Score**. Он одинаковый для 3 и 5 (0,82).

По **методу локтя** получаем, что 5 лучше подходит, чем 3.

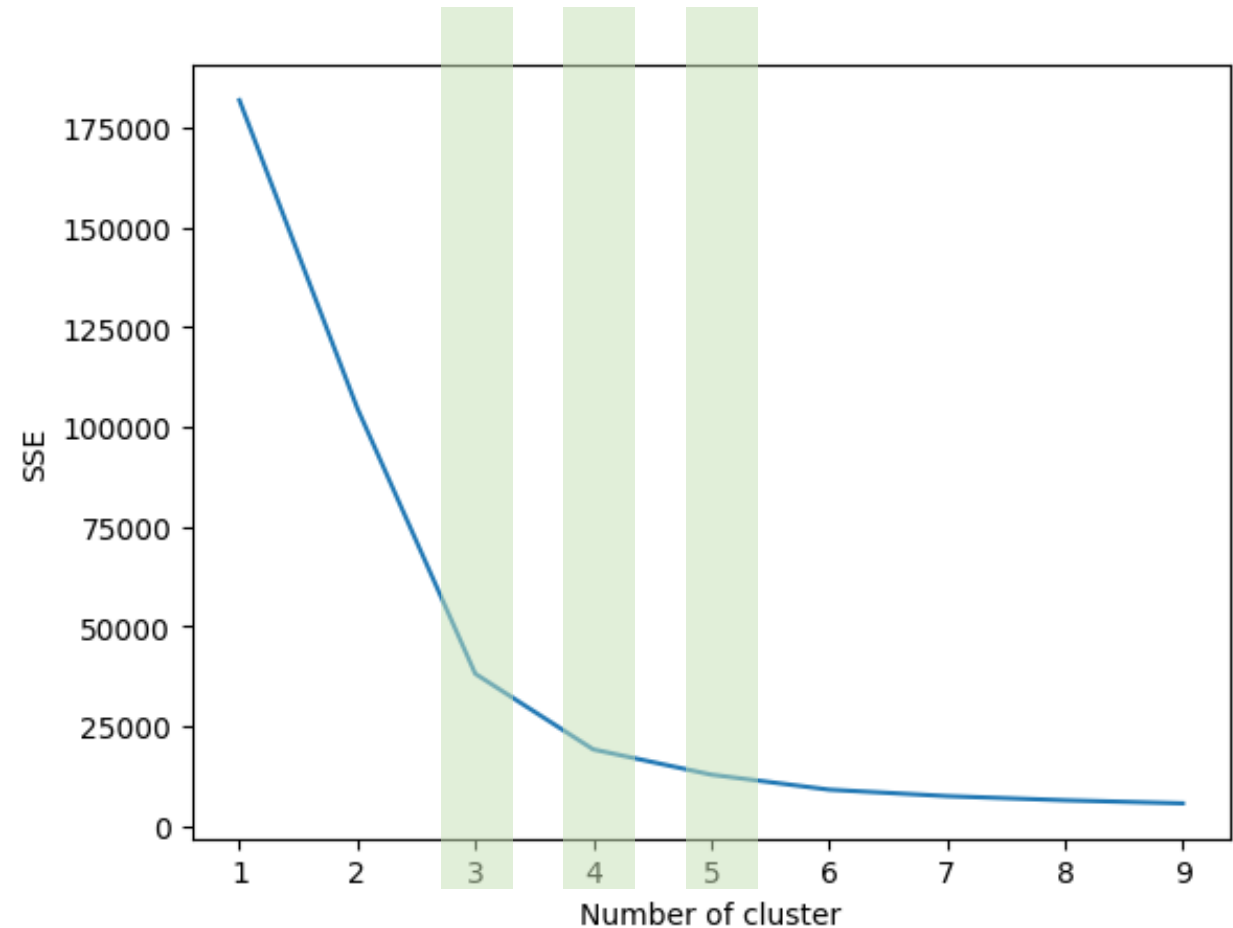


График SSE (расстояние от точек до центра кластеров) от k для кластеризации методом K-Means



Кластеризация, KMeans

Тогда проведем визуальный анализ.
Видим, что 5 лучше делит данные.

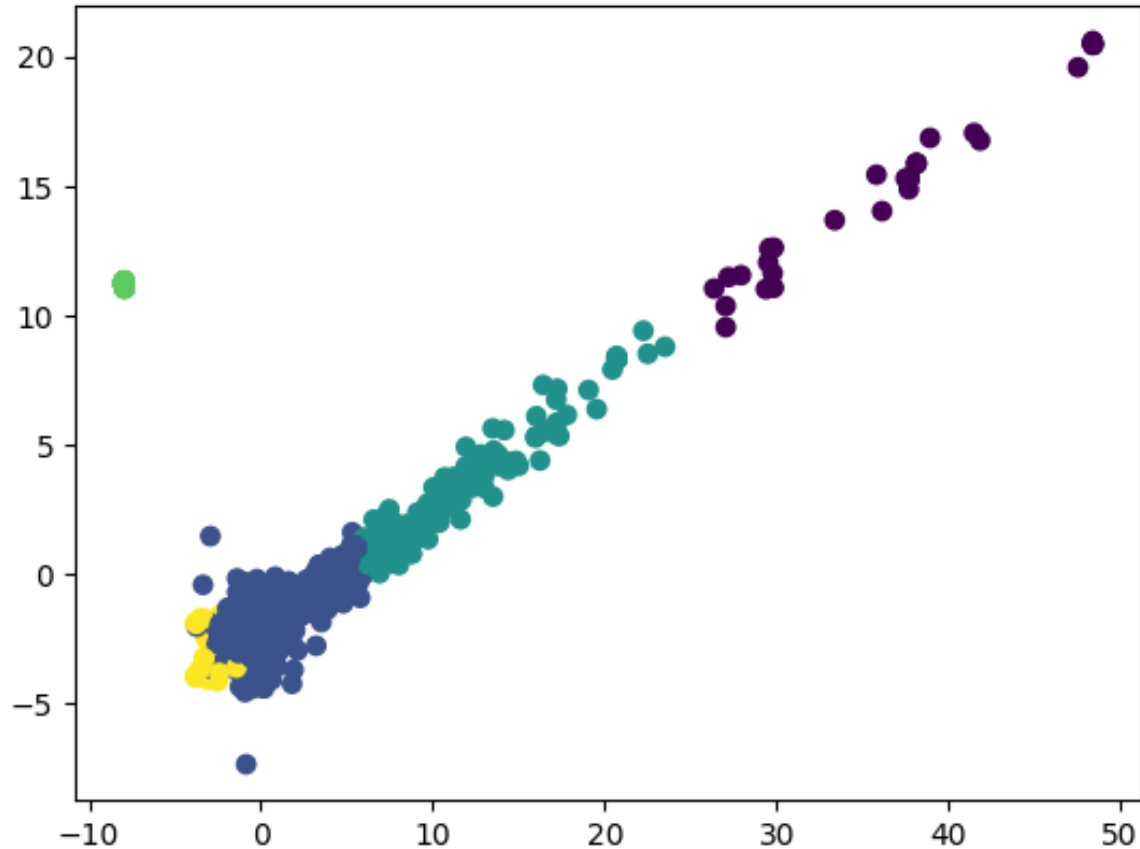


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.

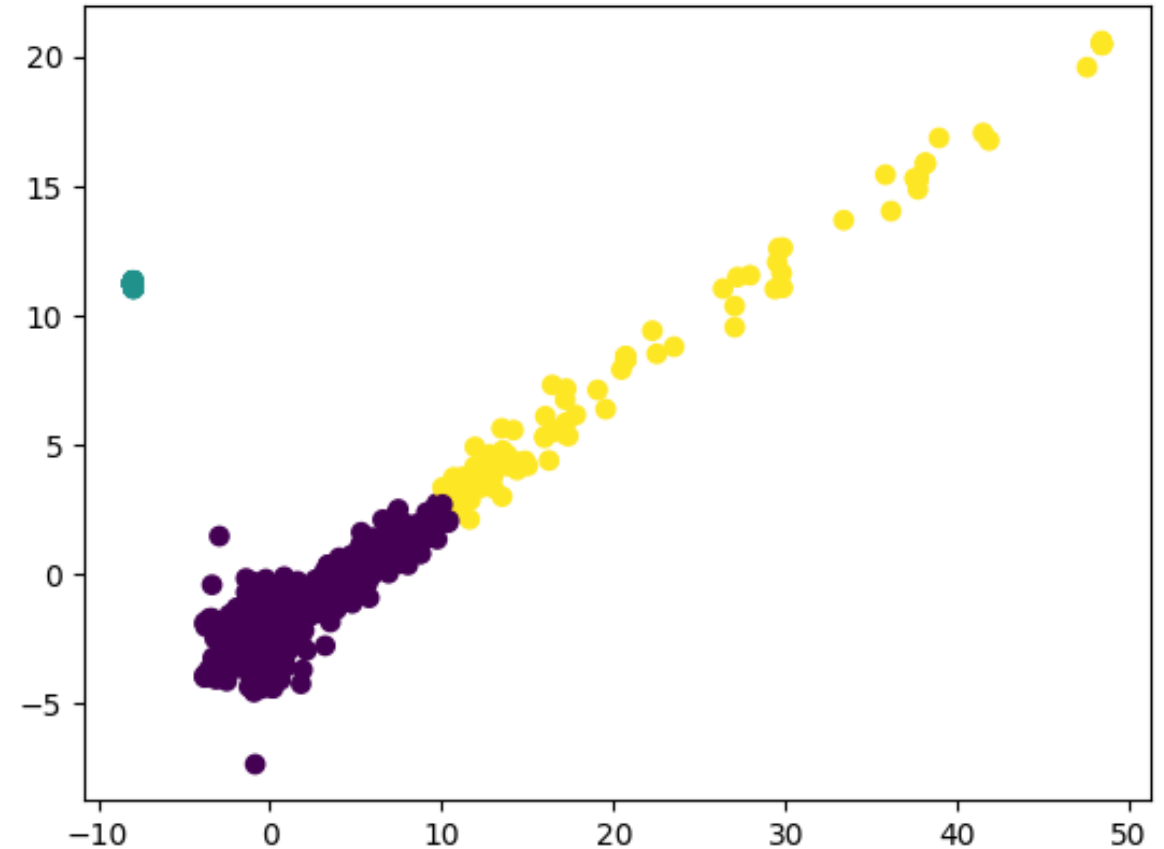


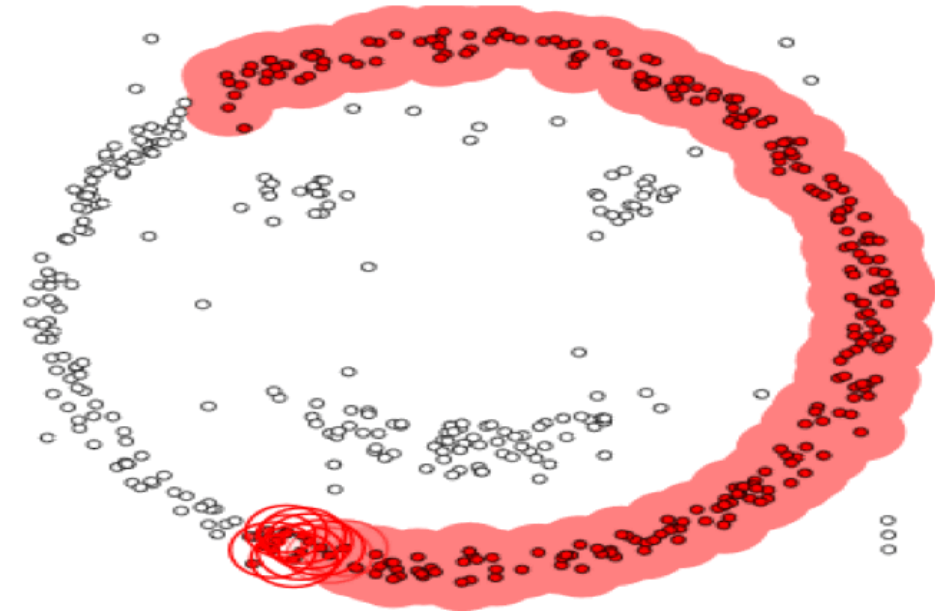
График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 3.

Кластеризация, DBSCAN

Алгоритм DBSCAN. Данный алгоритм принимает на вход параметр **eps**, который определяет минимальное расстояние между двумя точками для того, чтобы они могли считаться соседями, а также параметр **minPts** — минимальное количество соседей в радиусе eps.

Алгоритм предполагает, что множество точек можно разбить на 3 типа: ядерные точки (они имеют более, чем minPts точек в шаре радиуса eps с центром в данной точке), краевые точки (они имеют менее чем minPts точек в своем eps-шаре, но при этом содержатся в окрестности ядерной точки), выбросы (остальные точки).

DBSCAN для каждой точки p ищет ее окрестность $B_{eps}(p)$. Если $|B_{eps}(p)| \geq minPts$, то p — не выброс. Тогда все точки внутри шара мы считаем частями одного кластера.



Пример работы DBSCAN. Источник: <https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html>



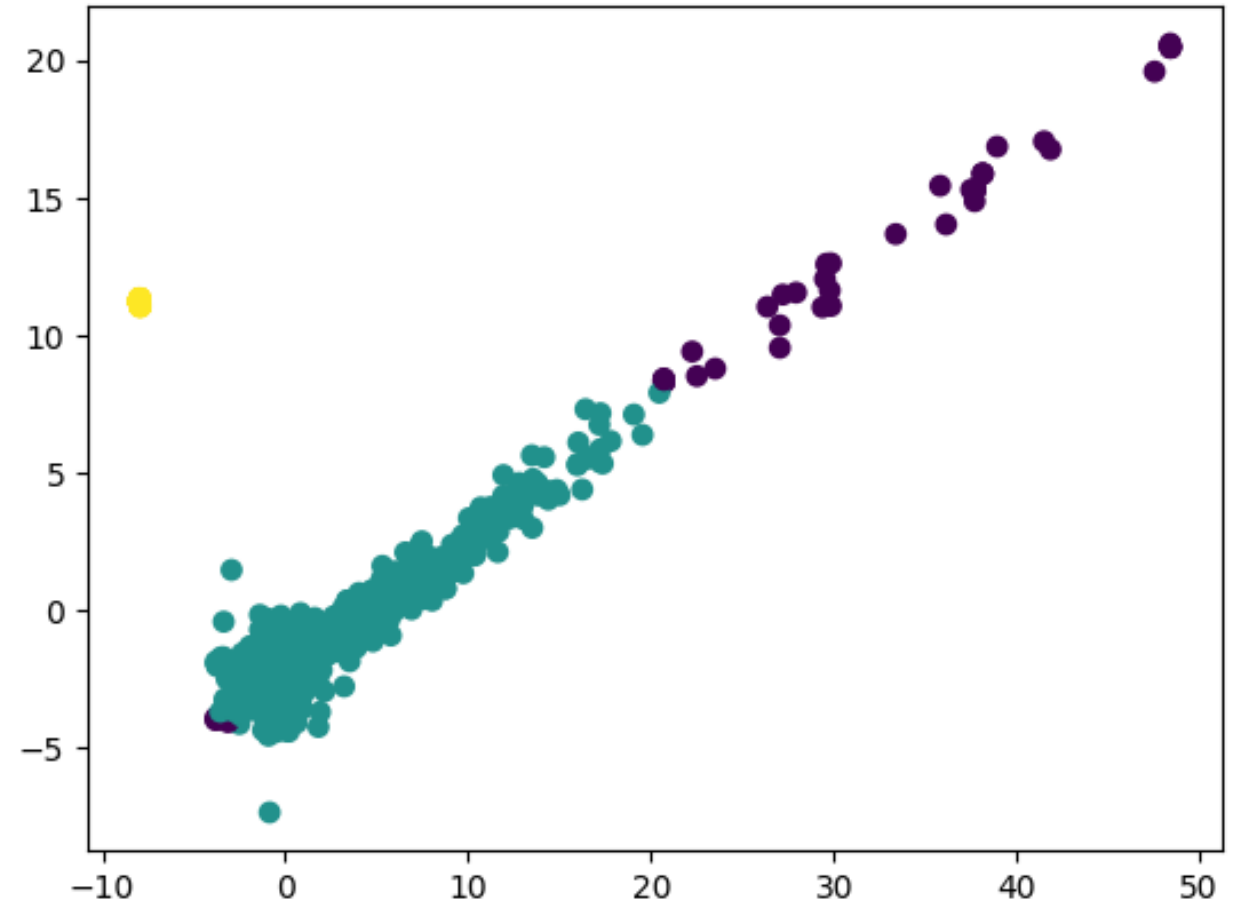
Кластеризация, DBSCAN

Предположим, что $eps = 4,95$ (это было подобрано эмпирически).

Подберем параметр $minPts$. Среди различных значений сильно выбивается значение Silhouette Score при $minPts = 27$ — 0,82.

Результат очень похож на K-Means по данной метрике. Данные очень плотные, поэтому DBSCAN не сработал лучше KMeans.

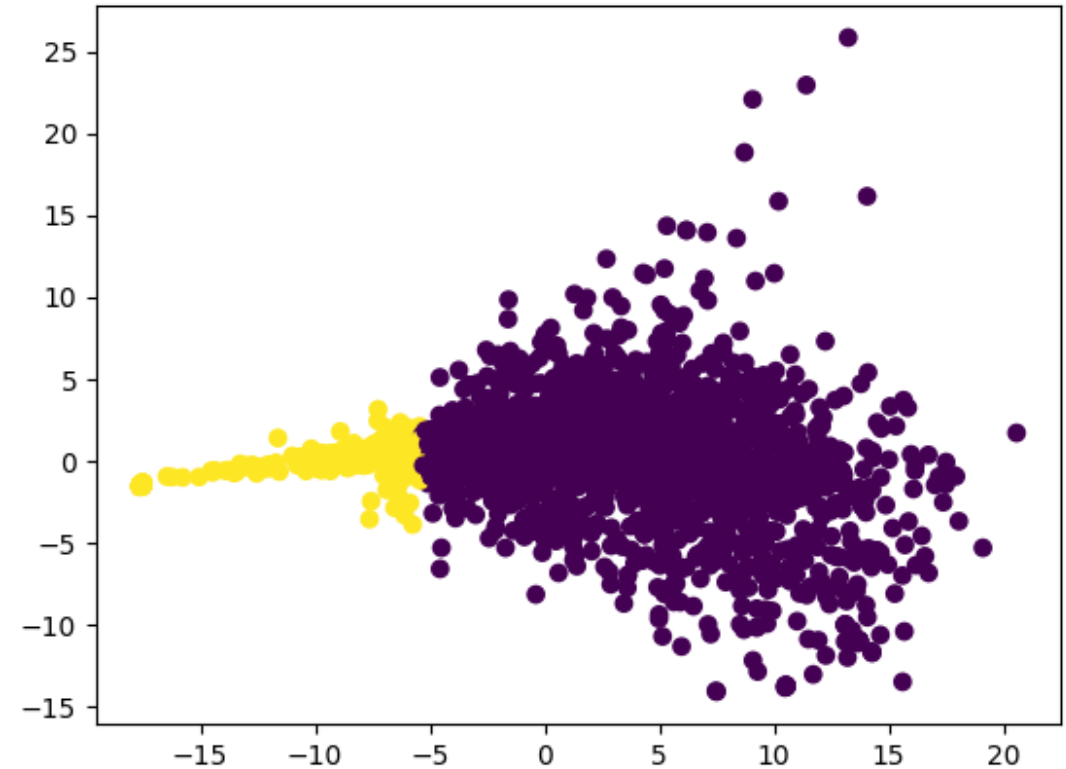
Итог: выбираем k-Means с $k = 5$.



Кластеризация методом DBSCAN на двумерной PCA-проекции
нормализованных данных

Анализ результатов

- Частичное обучение позволяет слегка улучшить метрики, полезные для практической применимости модели: F_1 -мера, количество нулей среди предсказаний.
- При этом в работе были проведены три эксперимента: на 4%, 10% и 14% размеченных данных.
- При переходе от меньшей доли размеченных данных к большей метрики резко повышались, демонстрируя скачок, равный до 50% от своего изначального значения.
- Таким образом, для качества модели классификации играет роль в первую очередь доля размеченных данных.
- На 1-граммах KMeans имеет Silhouette Score на двух кластерах — он будет равен 0.553, сильно ниже. Таким образом, 3-граммы действительно улучшают качество кластеризации.



Кластеризация методом KMeans на два кластера при
использовании 1-грамм



Сегментация и текстовая аналитика

Как приложить модель? Например, чат-бот. Или прогнозирование при долгосрочном планировании.

Предположим, что в отделе аналитики некоторой компании активно ведется работа над n_1 крупными проектами и n_2 небольшими проектами.

В компании работает m_1 опытных аналитиков и m_2 начинающих аналитиков. Чтобы повысить производительность сотрудников, необходимо нанять еще больше аналитиков.

Для этого требуется провести маркетинговую компанию и прорекламирровать открытые позиции начинающих и опытных аналитиков.

Осуществим сегментацию методом K-means с $k = 5$.

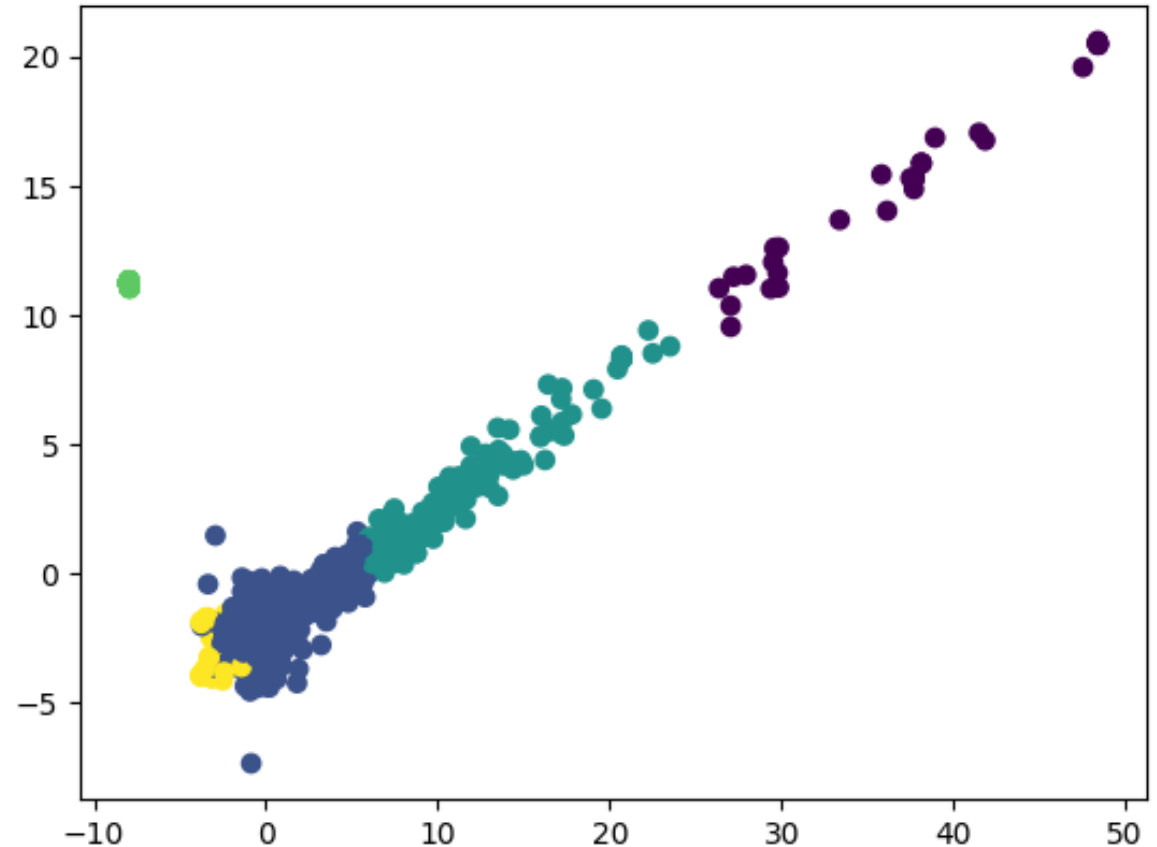


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.



Контекст триграмм

Мы сильно сократили описания из раздела “О себе”, поэтому нам необходимо научиться работать с контекстом.

Первая идея: x и y имеют общий контекст, если y входит в топ близких к x по word2vec триграмм. Возникает проблема.

Рассмотрим триграммы, похожие на “**нестандартный подход решение**” по word2vec. Самая похожая — “**ум способность работать**”, контекст общий.

Однако уже 4 по сходству — “**письменный речь быстрый**”. Она не имеет прямой контекстуальной связи с триграммой “**нестандартный подход решение**”.

Решение проблемы — по аналогии с графами, смоделируем тернарное отношение слова в нормальной форме (x, y, z) лежат в одном контексте с помощью симплициальных комплексов.



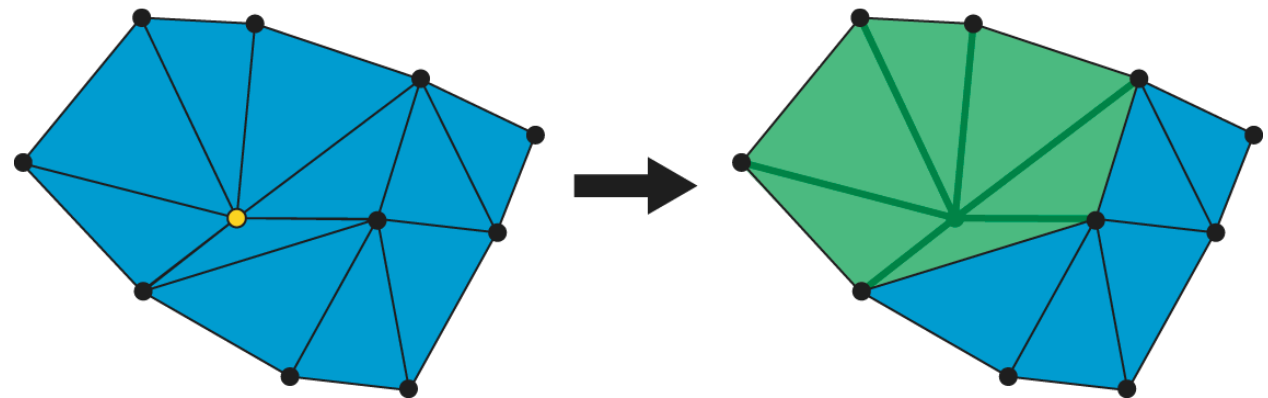
Пример контекста через word2vec. Источник: <https://stackoverflow.com/questions/65852710/text-similarity-using-word2vec>

Контекст триграмм

Симплициальный комплекс \mathcal{K} — это множество симплексов S таких, что каждое подмножество $S' \subset S$ каждого симплекса содержится в комплексе, а также любые два симплекса $S, S' \in \mathcal{K}$, которые имеют непустое пересечение, содержат $S \cap S'$ в качестве своих подмножеств. Размерность симплекса S определяется как $\dim S = \#S - 1$.

Пусть \mathcal{K} — симплициальный комплекс, \mathcal{S} — набор симплексов данного комплекса. Тогда определим унарную операцию звезды набора симплексов \mathcal{S}^* как объединение звезд S^* для каждого симплекса $S \in \mathcal{S}$. Для отдельного симплекса S операция звезды S^* определяется как набор симплексов S' , содержащих S в качестве подмножества.

См. Графический пример. Точки (0-симплексы) — слова и т. д. Тогда операция звезды над точкой (словом) вернет все биграммы и триграммы, в которых встречается это слово. Данная операция позволяет смоделировать контекст триграмм. Удобство: переход между размерностями отношений.



Пример операции звезды для желтой точки, источник <https://math.stackexchange.com/questions/633307/definition-of-star-in-a-simplicial-complex#633355>



Интерпретация сегментов

Большая часть резюме, 2130 штук. **9% было отнесено к 1, 4,4% — ко 2 классу.** Большое число **опытных специалистов**, имеющих опыт работы в IT на позиции разработчика или аналитика, понимают задачи автоматизации бизнес-процессов и внедрения систем автоматизации в работу бизнеса, обладают развитыми soft skills.

Зачастую умеют работать с иностранными клиентами, понимают, как оптимизировать бизнес-процессы. Триграммы “разработка программный обеспечение”, “крупный российский it”, “заниматься информационный технология”, “настоящий время работать”, “настоящий время заниматься”, “иметь опыт работа”, “опыт работа сфера”

Контекст: “лидер российский it”, “бизнес автоматизация внедрение”, “анализ бизнес оптимизация”, “сфера работать аналитик”, “навык ведение переговоры”, “клиент переговоры иностранный”

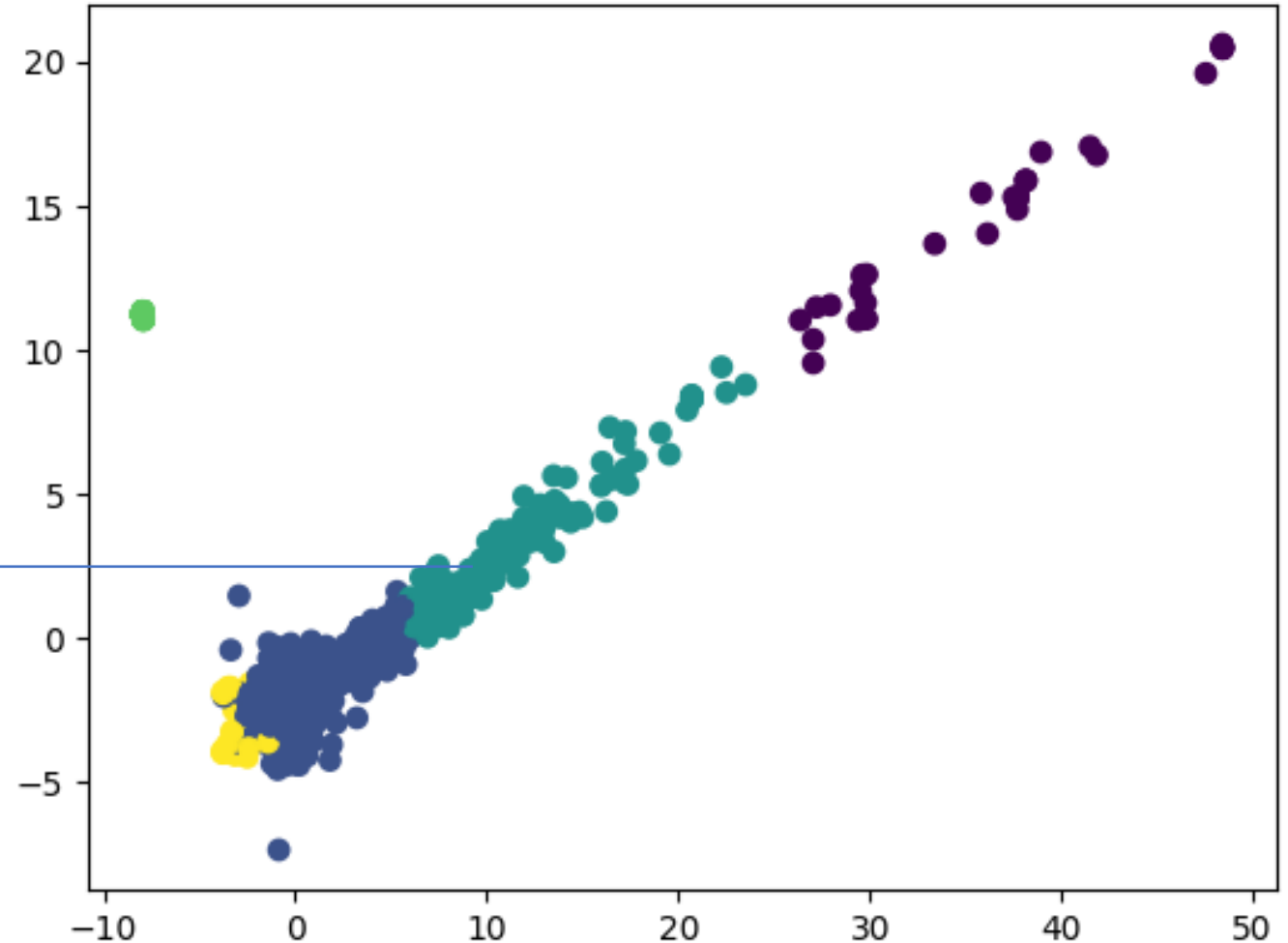


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.



Интерпретация сегментов

Во втором кластере содержится 356 резюме с пустым разделом “О себе”, поэтому они не поддаются анализу в рамках текстовой аналитики.

При этом с вероятностью 1% модель выдает оценку “2” и с вероятностью 4% — оценку “1”.

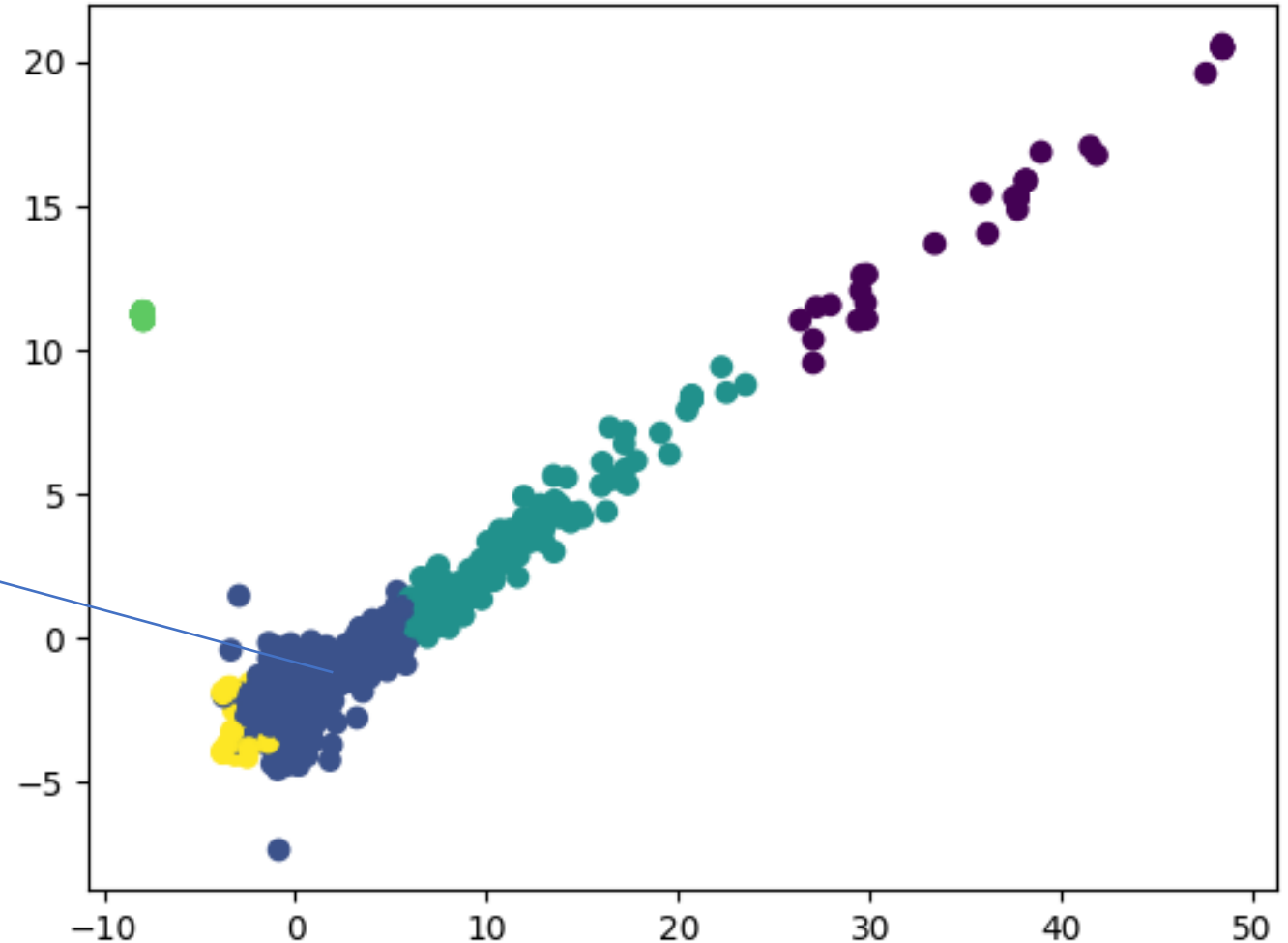


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.

Интерпретация сегментов

3 кластер — 37 резюме, 0 — “2”, а 22 резюме — “1”. Триграммы: “работать сфера маркетинг”, “сфера маркетинг реклама”, “работать сфера торговля”, “работать сфера продажа”, “опыт работа пк”.

Контекст: “ms работа опыт”, “excel работа опыт”, “point работа опыт”, “консультант работа опыт”, “работать маркетинг партизанский”, “работать коллектив любить”.

Не IT, знают только базовые инструменты (MS Office, КонсультантПлюс), занимались не самыми актуальными задачами (напр., партизанский маркетинг), не имеют впечатляющего резюме, поэтому в качестве soft skills описывают любовь к коллективу.

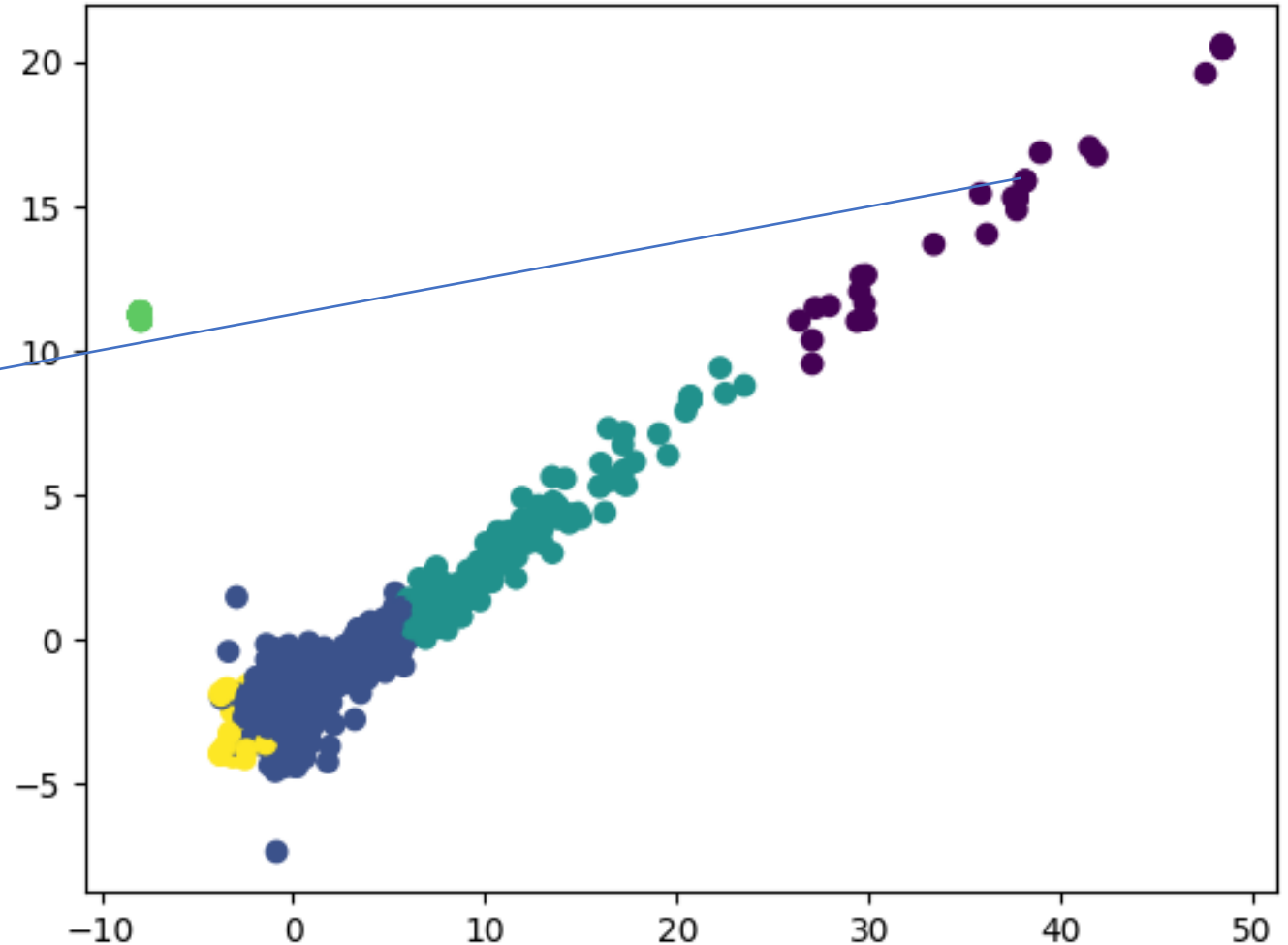


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.

Интерпретация сегментов

Мы видим крайне однообразные триграммы: “данный момент работать”, “находиться поиск работа”, “работа данный момент”, “поиск работа хотеть”. По всей видимости, это кластер людей, которые в разделе “О себе” не написали почти ничего, кроме того, что они хотят работы.

Пример: “Хорошо работаю в команде, умею решать конфликты и идти на компромисс. С энтузиазмом берусь за новые проекты и довожу дело до конца. Обладаю аналитическим складом ума, быстро генерирую необычные идеи для решения задач пользуясь творческим подходом к делу.”

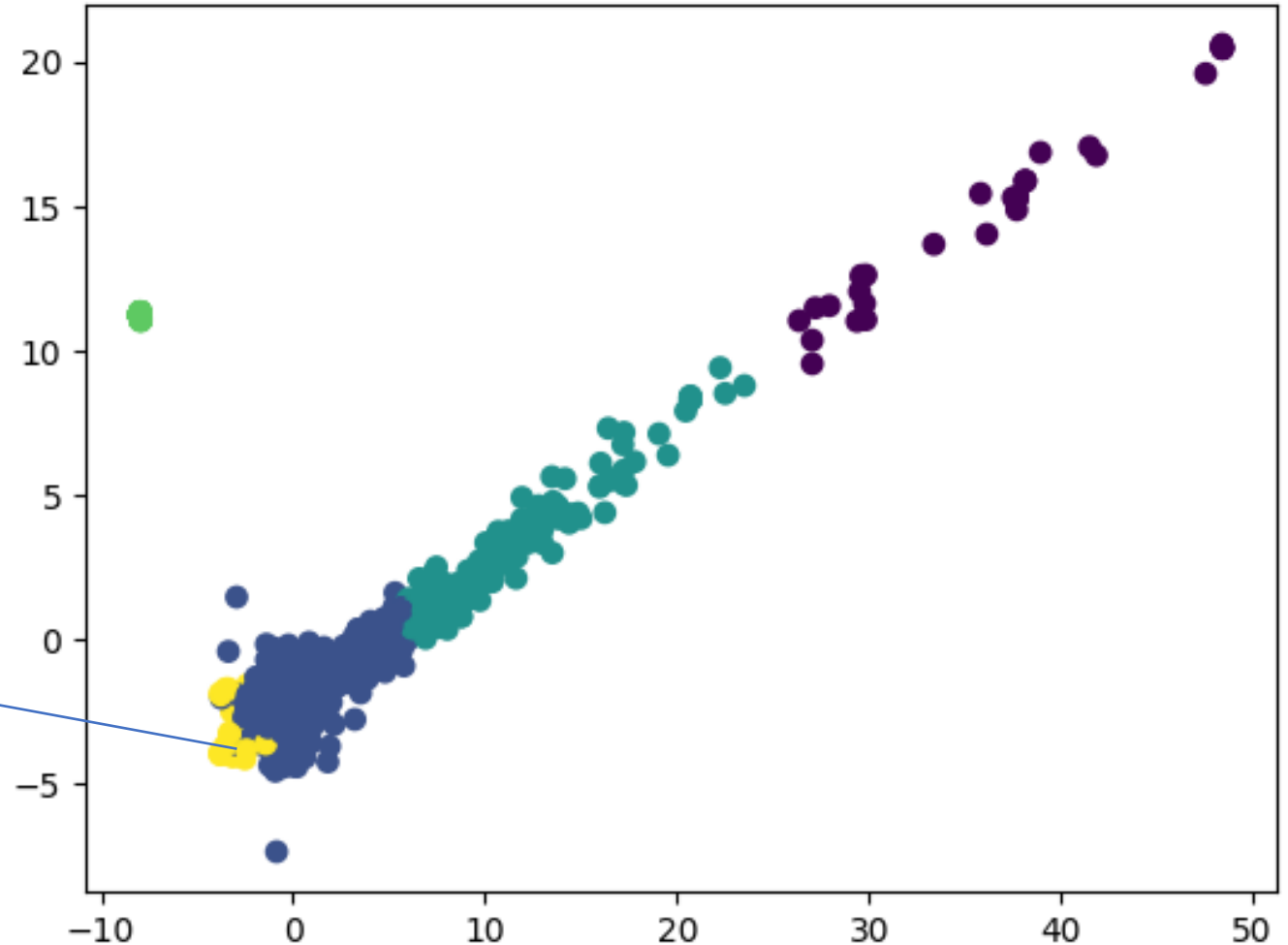


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.



Интерпретация сегментов

Совсем пустые описания. Пример описания:
“Интиллегентный, ответственный, целеустремлённый.
Аналитик по натуре. Готов к командировкам,
самоорганизован.”

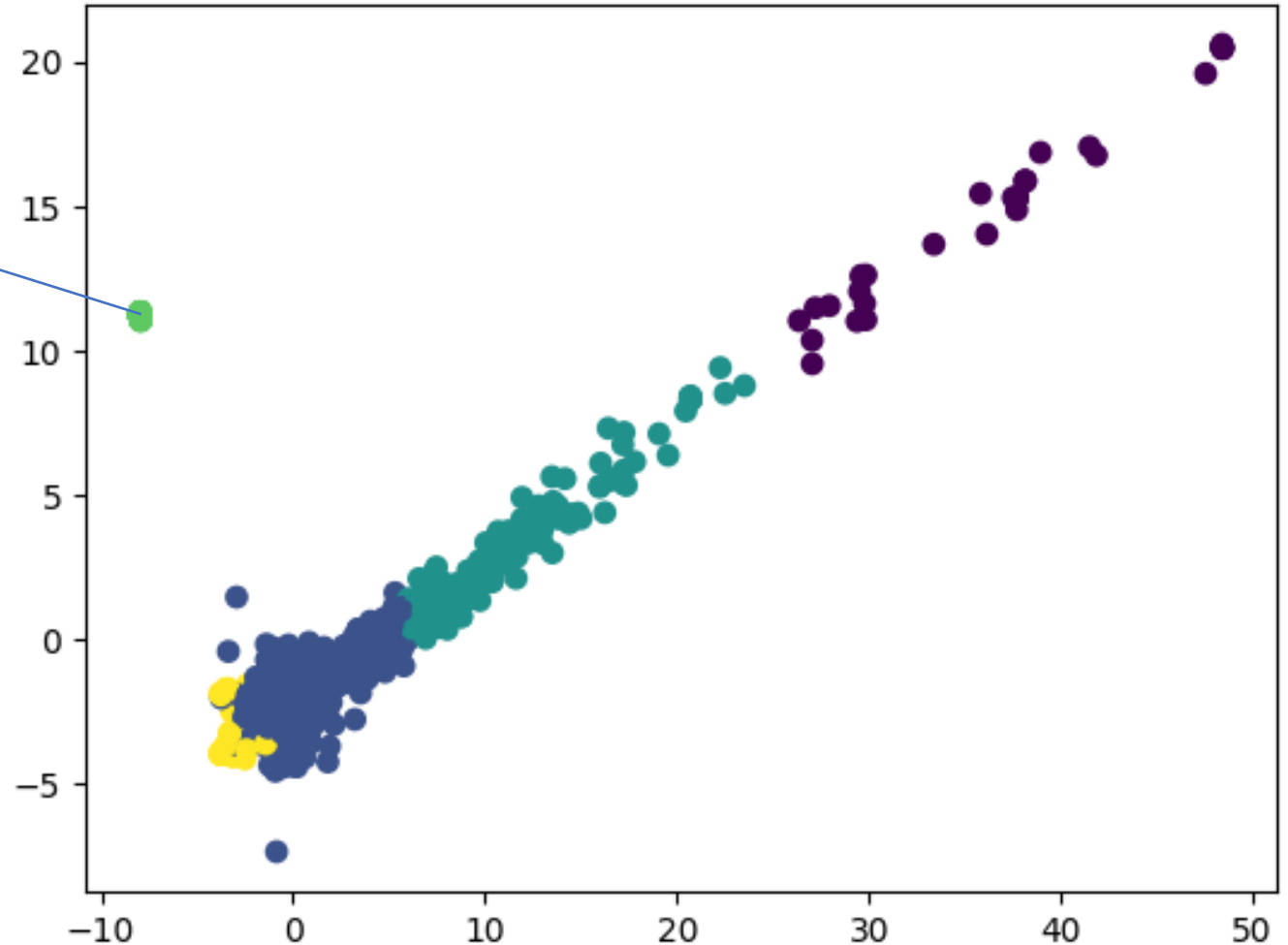


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.

Интерпретация сегментов

Ясно, что лучше всего выбирать людей из первого кластера. Первая группа. Наиболее популярные триграммы: “настоящий время работать”, “время работать крупный”, “иметь опыт работа”, “требоваться высокопоставленный сотрудник”, “самый известный российский”, “работоспособность вариативный мышление”.

Данные триграммы можно проинтерпретировать так: опытный кандидат прямо сейчас работает в крупной компании, но ищет еще более выгодную высококвалифицированную должность, хочет выполнять сложные и творческие задачи.

Контекст: “российский известный вуз”, “компания российский известный”, “высокий коммуникативный работоспособность”, “лидер нацелить результат”, “самостоятельно решение принимать”.

То есть, кандидат имеет очень хорошее резюме, имеет лидерские и организационные качества — вполне возможно, что его можно даже нанять на руководящую позицию.

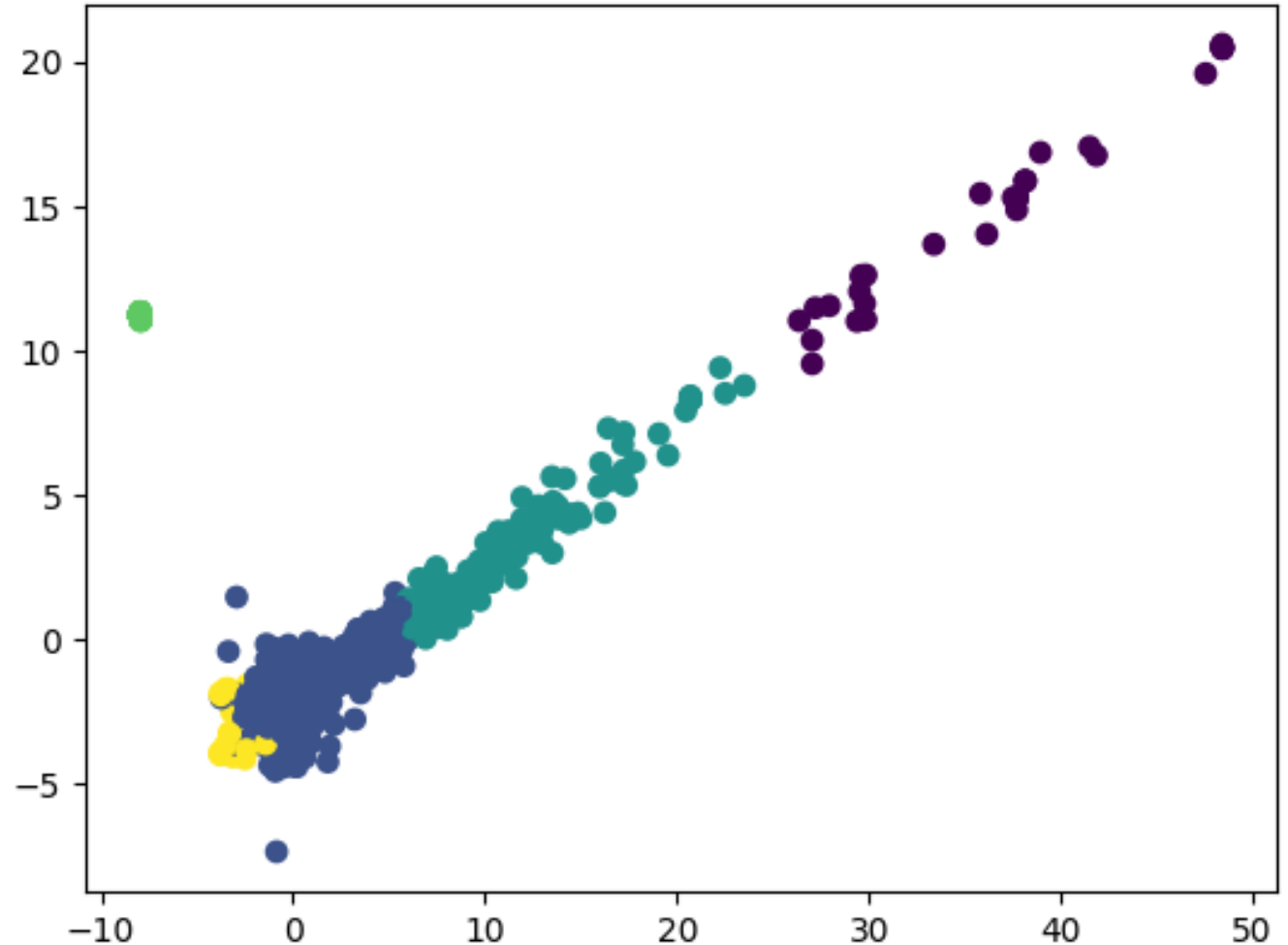


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.

Интерпретация сегментов

Вторая группа. Триграммы: “находиться поиск работа”, “данный момент работать”, “иметь опыт работ”, “разработчик программный обеспечение”, “большой объём информации”, “направление анализ данные”, “моделирование бизнес процесс”.

Мы видим, что это группа людей, которые обладают неплохие hard skills, работали в сфере, где акцент делается на программировании, моделировании бизнес-процессов и разработке, уже имеют опыт работы, где занимались, скорее всего, рутинными задачами для начинающих, в том числе с большим объемом данных и анализом данных, однако, скорее всего, хотят переквалифицироваться.

Контекст: “бизнес бд моделирование”, “бизнес автоматизация моделирование”, “большой объём собирать”, “большой объём анализировать”, “разработчик программный интерфейс”.

Мы видим, что, действительно, навыки связаны с тем, что необходимо аналитику (анализ данных, работа с базами данных), но предыдущий опыт работы был связан с более рутинными инженерными задачами.

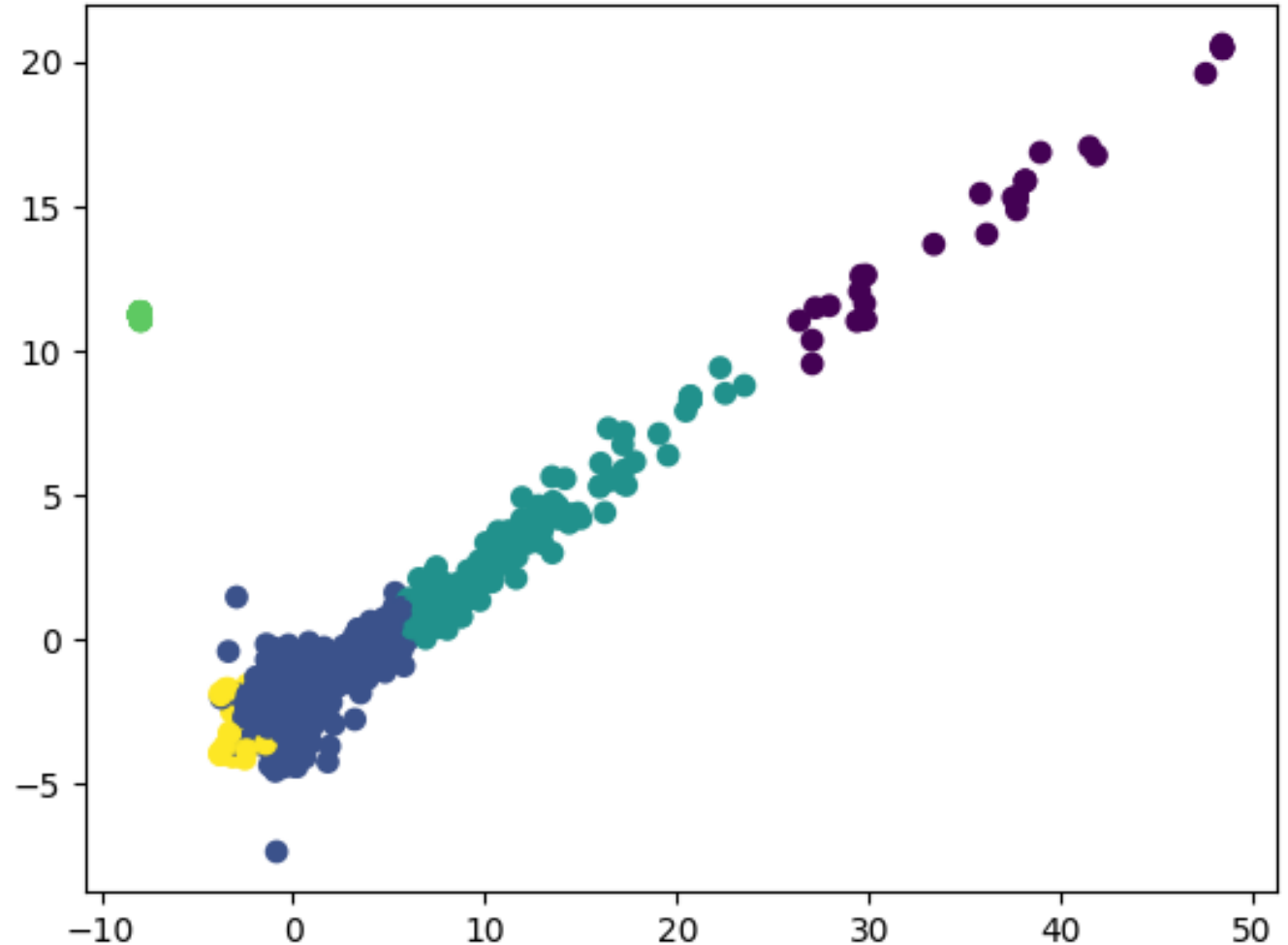


График кластеров, полученных алгоритмом K-Means, проекции нормализованных данных, полученных с помощью PCA, при количестве кластеров 5.



Прогнозирование и интерпретируемость

Предположим, что на рекламу откликается $p(seen)$ человек и что среди тех кандидатов в кластере, кто увидел рекламу, столько же, сколько среди всех кандидатов вообще в данной группе: $p(seen | clust1) = p(seen) =: p$

Тогда $len(clust1) \cdot p(y = x | seen, clust1) = N_x, x \in \{1, 2\}$ — сколько посмотрит рекламу среди различных групп.

Мы уже заметили, что $p(y = 1 | clust1) \approx 9\%$, $p(y = 2 | clust1) \approx 4,4\%$. Тогда собеседование пройдет $p \cdot N_1 \approx p \cdot 0.09 \cdot 2130 \approx 191p$ человек с оценкой 1 и $93.72p$ с оценкой 2. Предположим, что $p \approx 10\%$.

Тогда мы получим примерно 9 человек на сложные проекты и 19 человек на простые, что может быть вполне приемлемо и окупать маркетинговую компанию.

Чем больше информации содержится в кластере, тем более он интерпретируемый. Посчитаем между парами различных токенов из топ-10 их среднее сходство по word2vec. Чем более сходство, тем хуже интерпретируемость. И она повысилась, что можно заметить в таблице справа.

Топ 1-грамм в наилучшем кластере	Топ 3-грамм в наилучшем кластере
работа	настоящий время работать
опыт	гражданин россия проживать
компания	уверенный пользователь пк
россия	умение работать команда

Топ 1-грамм и 3-грамм в наилучших кластерах при различение текстов на 1-граммы и 3-граммы соответственно



Заключение

В итоге была получена таблица с прогнозами. Таким образом, результатами алгоритма сможет воспользоваться и аналитик, и менеджер.

Традиционные методы: обучение с учителем для двухклассовой классификации (метод опорных векторов, логистическая регрессия и другие алгоритмы) и тематическое моделирование (алгоритмы LDA, LSA и другие).

В нашей работе мы попытались усовершенствовать классические методы классификации и тематического моделирования, обучив модель частичного

обучения self-train для трех-классовой классификации и кластеризуя триграммы с помощью различных методов кластеризации.

№	Опыт	Топовость образования	Релокация?	Город?	...	DA
0	7.08	Не входит в топ	Готова к редким командировкам	Москва	...	0.0
9	2.5	Не входит в топ	Не готов	Москва	...	1.0
41	2.9	Входит в топ	Готов	Москва	...	1.0

Таблица с данными после применения алгоритмов классификации



Заключение

В результате мы выяснили, что специалистов по Data Science нельзя размечать вместе с аналитиками, что триграммы сильно увеличивают качество кластеризации и интерпретируемость кластеров, частичное обучение дает небольшое улучшение метрик, влияющих на практическую применимость модели, — F_1 -меру, долю нулей среди ответов модели. При этом было выяснено, что в первую очередь на качество классификации влияет количество размеченных данных.

Помимо этого, в работе было рассмотрено приложение построенных моделей в задаче сегментации резюме для долгосрочного принятия решений. В этой задаче мы разработали новое решение на базе симплициальных комплексов, рассмотрели приложения модели классификации, проинтерпретировали кластеры.

В результате в данной работе мы исследовали задачу автоматизации принятия решений в сфере HR и разработали методы улучшения качества кластеризации и интерпретируемости кластеров, а также улучшения качества классификации резюме и возможности для обучения нескольких целевых признаков на одном наборе данных, после чего рассмотрели приложения решенных задач для задачи сегментации резюме.



Заключение

Что можно улучшить:

1. Из неравенства на норму, которое мы доказали в главе 2.3, следует, что средний word2vec-вектор будет завышен для похожих между собой текстов. К примеру, в таблице 3.4 видно, что для практически идентичных резюме $d(x, y) > 0.3$. Необходимо модифицировать данный метод нахождения расстояния между текстами. К примеру, для похожих векторов высчитывать его как $d(x, y) - \varepsilon$ для некоторого $\varepsilon > 0$, чтобы компенсировать завышение. Найти ε можно, к примеру, на основе неравенств на нормы из функционального анализа или статистически
2. Рассмотреть более сложные алгоритмы для симплициальных комплексов и применить их для нахождения контекста. К примеру, можно рассмотреть симплициальные комплексы с (x, y) в качестве весов, где (x, y) — скалярное произведение между триграммами x и y или среднее скалярное произведение между векторами, содержащими x, y , если x, y — биграммы.
3. Разметить больше единиц и двоек и посмотреть на качество алгоритма после этого
4. Рассмотреть более сложные модели принятия решений, более сложные алгоритмы машинного обучения

