Разработка гибридного метода вероятностного поиска и верификации записей в базах данных

Гильмутдинов Тимур Артурович, Магистрант, Московский физико-технический институт (национальный исследовательский университет), Долгопрудный, Россия
Подлесных Дмитрий Артурович, Аспирант, Московский физико-технический институт (национальный исследовательский университет), Долгопрудный, Россия

Цель исследования – разработка гибридного метода вероятностного поиска и верификации записей в реляционных базах данных, позволяющего сопоставлять строковые атрибуты, содержащие опечатки, сокращения и разночтения. Методология базируется на вычислении нормированного расстояния Левенштейна для каждого атрибута, формировании байесовской оценки апостериорной вероятности совпадения записи из внешнего источника с эталонной записью и выборе кандидата, максимизирующего эту вероятность. Для повышения производительности применяются C-расширения (библиотека rapidfuzz), ограничение числа кандидатов топ-20 наиболее похожих записей по ФИО, векторизация операций с помощью pandas и предварительная нормализация строк. Ключевые результаты: разработанный метод демонстрирует устойчивость к искажениям входных данных, учитывает одновременно несколько атрибутов (ФИО, идентификаторы, наименования) и не требует размеченной обучающей выборки, в отличие от классического вероятностного подхода Fellegi–Sunter. Экспериментальная проверка на эталонной базе из 250 тысяч записей и тестовой выборке из 159 искажённых записей показала время обработки 62 секунды на стандартном оборудовании. Вывод: предложенный гибридный метод сочетает гибкость нечёткого сравнения строк и вероятностную оценку, что делает его эффективным для задач интеграции и очистки данных в корпоративных информационных системах.

Введение

Проблема согласования разнородных данных возникла задолго до появления современных корпоративных систем. В 1960–1970-х годах, с развитием реляционной модели данных [E.F. Codd, 1970], стало очевидно, что даже при строгой схеме данных реальные операционные системы неизбежно порождают опечатки, сокращения и различные форматы записи одних и тех же сущностей. Это привело к формированию направлений entity resolution (разрешение сущностей), record linkage (связывание записей) и data deduplication (дедупликация), активно развивающихся с 1980-х годов в работах I.P. Fellegi, A.B. Sunter, W.E. Winkler и др. [1, 2].

Важным этапом эволюции методов интеграции стала стандартизация форматов обмена и семантического описания. В 1980–1990-х годах для межкорпоративного обмена в логистике и финансах начали применяться стандарты EDIFACT и SWIFT, которые регламентировали строгие структуры сообщений, но оставляли проблему нечёткого сопоставления строковых атрибутов на уровне прикладных систем. С развитием веб-технологий и концепции Semantic Web [3] возникли стандарты RDF, OWL, а затем инициатива Linked Data [4]. Организации OASIS разработали спецификации UBL, а консорциум RosettaNet предложил строгие словари для высокотехнологичных отраслей [5]. Несмотря на формализацию семантики, на практике значительная часть данных продолжает поступать в виде неструктурированных или слабоструктурированных текстовых полей, что возвращает исследователей к задачам вероятностного сопоставления.

В настоящий момент одной из ключевых проблем при интеграции и очистке данных в корпоративных информационных системах является наличие искажений и разночтений в строковых атрибутах датафреймов. К таким искажениям относятся опечатки в ФИО, различные формы написания идентификационных номеров (СНИЛС, ИНН), сокращения в наименованиях организаций. Классические детерминированные методы (функция ВПР в MS Excel, метод merge в pandas [10]) показывают низкую полноту и требуют точных совпадений. Поэтому необходимо рассмотрение методов нечёткого (вероятностного) поиска [6, 7].

Методология: расстояние Левенштейна и нормированная оценка схожести

Одним из наиболее распространённых подходов к оценке схожести строк является алгоритм Левенштейна (редакционное расстояние). Данная метрика определяет минимальное количество операций вставки, удаления и замены символов, необходимых для преобразования одной строки в другую. Рекуррентная формула для вычисления расстояния $D_{(i,\ j)}$ между S_1 и S_2 имеет вид:

$$D_{(i,\ j)}=\begin{cases} 0;i=0,\ j=0 \\ i;j=0,\ i>0 \\ j;i=0,\ j>0 \\ min\begin{pmatrix} D_{(i,\ j-1)}+1 \\ D_{(i-1,\ j)}+1 \\ D_{(i-1,\ j-1)}+m(S_1[i],\ S_2[j]) \end{pmatrix};i>0,\ j>0 \end{cases}, \quad (1)$$

где $S_1,\ S_2$ – исходные строки;

$D_{(i,\ j)}$ – элемент матрицы расстояний для первых i символов S_1 и первых j символов S_2;

$m(a,\ b)$ – функция стоимости замены: 0 если символы равны, иначе 1.

Для получения оценки схожести двух строк в процентах применяется нормированное отношение, реализованное в Python-библиотеке rapidfuzz [8]:

$$ratio(S_1,\ S_2)=\frac{(len(S_1)+len(S_2)-D(S_1,S_2))*100}{(len(S_1)+len(S_2))}, \quad (2)$$

где $ratio(a,\ b)$ – функция определения степени схожести строк: 100% если они идентичны;

$S_1,\ S_2$ – исходные строки;

$len(a)$ – функция извлечения длинны строки a.

Значение 100 % соответствует полной идентичности строк.

Байесовская постановка задачи поиска

Предлагается при сопоставлении искажённой записи с эталонной базой данных сравнить записи по нескольким критериям, используя вероятностный подход [9]. Пусть T – искаженная запись из внешнего файла, содержащего атрибуты a, b, c, …, а S – кандидат на совпадение из основной базы данных. Необходимо найти такую запись $\hat{S}$, которая максимизирует условную вероятность $P(S|T)$. Согласно формуле Байеса:

$$\hat{S}=\ max_s P(S|T)=\ max_s P(S)\cdot P(T|S), \quad (3)$$

где $P(S)$ – априорная вероятность записи в базе,

$P(T|S)$ – вероятность того, что эталонная запись S могла быть искажена до вида T.

В данной работе предлагается аппроксимировать $P(T|S)$ как произведение независимых вероятностей совпадения по N ключевым атрибутам:

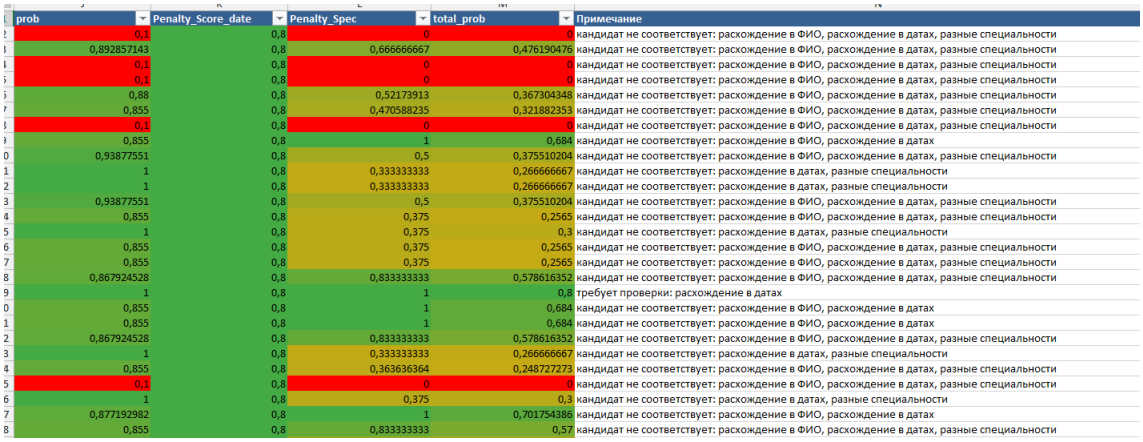
$$P(T|S)=\ \prod_{k=1}^{N} ratio\_norm_k(T_k,S_k\ ), \quad (4)$$

где $ratio\_norm_k(a,b\ )$ – функция схожести строк *a*, *b*, нормированная в диапазон [0, 1].

Таким образом формируется модель, которая снижает итоговую вероятность при несоответствии записей и выбирает запись с максимальной вероятностью совпадения.

Реализация и оптимизация

На основе вычисления вероятностей совпадения атрибутов реализована рекомендательная система на основе конечных автоматов, которая возвращает текстовые примечания для коррекции действий пользователя информационной системы (рис. 1).



prob	Penalty_Score_date	Penalty_Spec	total_prob	Примечание
0,1	0,8	0	0	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,892857143	0,8	0,666666667	0,476190476	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,1	0,8	0	0	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,1	0,8	0	0	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,88	0,8	0,52173913	0,367304348	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,855	0,8	0,470588235	0,321882353	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,1	0,8	0	0	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,855	0,8	1	0,684	кандидат не соответствует: расхождение в ФИО, расхождение в датах
0,93877551	0,8	0,5	0,375510204	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
1	0,8	0,333333333	0,266666667	кандидат не соответствует: расхождение в датах, разные специальности
1	0,8	0,333333333	0,266666667	кандидат не соответствует: расхождение в датах, разные специальности
0,93877551	0,8	0,5	0,375510204	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,855	0,8	0,375	0,2565	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
1	0,8	0,375	0,3	кандидат не соответствует: расхождение в датах, разные специальности
0,855	0,8	0,375	0,2565	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,855	0,8	0,375	0,2565	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,867924528	0,8	0,833333333	0,578616352	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
1	0,8	1	0,8	требует проверки: расхождение в датах
0,855	0,8	1	0,684	кандидат не соответствует: расхождение в ФИО, расхождение в датах
0,855	0,8	1	0,684	кандидат не соответствует: расхождение в ФИО, расхождение в датах
0,867924528	0,8	0,833333333	0,578616352	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
1	0,8	0,333333333	0,266666667	кандидат не соответствует: расхождение в датах, разные специальности
0,855	0,8	0,363636364	0,248727273	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
0,1	0,8	0	0	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности
1	0,8	0,375	0,3	кандидат не соответствует: расхождение в датах, разные специальности
0,877192982	0,8	1	0,701754386	кандидат не соответствует: расхождение в ФИО, расхождение в датах
0,855	0,8	0,833333333	0,57	кандидат не соответствует: расхождение в ФИО, расхождение в датах, разные специальности

Рис. 1. Работа рекомендательной системы по вероятностному поиску в базах данных

Для обеспечения приемлемой производительности на реальных объёмах данных применён ряд оптимизаций:

— С расширения: библиотека rapidfuzz реализует алгоритм Левенштейна на языке C, что ускоряет сравнение строк в 10–50 раз по сравнению с чисто Python реализациями.

— Ограничение кандидатов: для каждого ФИО ищутся только топ 20 похожих записей из эталонной базы; дальнейшие расчёты ведутся только по этому ограниченному набору.

— Векторизация: очистка строк и расчёт штрафов выполняются с помощью векторных операций pandas (apply и встроенные функции), что сокращает накладные расходы на интерпретатор Python [10].

— Предварительная обработка: нормализация всех строк выполняется один раз на этапе загрузки данных, а не в каждой итерации.

— Быстрая загрузка эталонной базы: данные хранятся в формате pickle, что позволяет загружать 250 000 записей за доли секунды.

Экспериментальные результаты

Для тестирования алгоритма была предоставлена эталонная база данных на 250 тысяч уникальных записей и искажённая выборка на 159 записей. На тестовом стенде (процессор Ryzen 9 7940HS, 16 ГБ RAM DDR5 4800 МГц) обработка заняла 1 минуту 2 секунды (рис. 2).

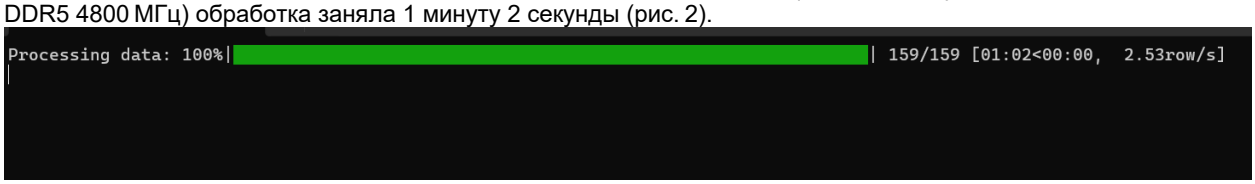


Рис. 2. Скорость обработки искажённого файла

В табл. 1 приведено сравнение ключевых подходов и стандартов, упомянутых в ретроспективе, с предлагаемым гибридным методом вероятностного поиска.

Таблица 1 - Сравнение ключевых подходов и стандартов с предложенной методикой

Технология / подход	Тип	Принцип работы	Преимущества	Ограничения
Классические детерминированные методы (ВПР, merge)	Детерминированный	Сравнение по точному совпадению одного или нескольких ключей	Простота, высокая скорость при идеально чистых данных	Не работают при опечатках, сокращениях, разночтениях
EDIFACT, SWIFT	Стандарты обмена	Строгие форматы сообщений, фиксированные идентификаторы	Обеспечивают интероперабельность в конкретных отраслях	Не решают проблему нечёткости внутри полей
RDF, OWL, Linked Data	Семантические технологии	Использование URI, онтологий, связывание данных через граф	Глубокая семантика, возможность логического вывода	Требуют предварительного моделирования, не рассчитаны на «грязные» текстовые данные
OASIS UBL, RosettaNet	Отраслевые словари и процессы	Стандартизированные бизнес-документы, коды, классификаторы	Унификация данных на уровне домена	Не учитывают искажения, возникающие при вводе
Классический вероятностный подход (Fellegi–Sunter)	Вероятностный	Сравнение по набору полей с весами, вычисляемыми на основе обучающих	Учитывает неоднородность качества полей, теоретически	Требует размеченных данных для настройки весов

		данных	обоснован	
Предлагаемый гибридный метод	Гибридный (вероятностный + редакционное расстояние)	Байесовская оценка по нескольким атрибутам, нормированные расстояния Левенштейна, ограниченный перебор кандидатов	Работает без обучающей выборки, устойчив к опечаткам, учитывает несколько полей одновременно, легко адаптируется	Производительность снижается при очень больших базах

Предлагаемый метод сочетает достоинства вероятностного подхода (учёт нескольких атрибутов) с гибкостью нечёткого сравнения строк (редакционное расстояние), что делает его эффективным в условиях реальных корпоративных данных, где искажения и разночтения неизбежны. В отличие от семантических стандартов (RDF, Linked Data), он не требует предварительного онтологического моделирования, а по сравнению с детерминированными методами обеспечивает высокую полноту поиска. Классический вероятностный подход Fellegi–Sunter требует размеченной обучающей выборки для оценки весов атрибутов, тогда как наша модель использует равномерные априорные вероятности и нормированные расстояния, что упрощает внедрение.

Основным ограничением остаётся производительность при работе с базами, содержащими миллионы записей: квадратичный перебор всех пар исключается за счёт ограничения кандидатов, но при очень широких таблицах может потребоваться дополнительная индексация (например, с использованием блочных методов). В будущем планируется исследовать возможность применения метрик, инвариантных к перестановкам слов, и интеграцию с векторными представлениями (embedding) для семантического сопоставления.

Заключение

Разработан гибридный метод вероятностного поиска и верификации записей, основанный на байесовской оценке с использованием нормированного расстояния Левенштейна. Метод не требует обучающей выборки, устойчив к опечаткам и сокращениям, учитывает несколько атрибутов одновременно. Экспериментально подтверждена его работоспособность на базе из 250 000 записей: обработка тестовой выборки заняла около 62 секунд. Предложенный подход может быть рекомендован для задач интеграции и очистки данных в корпоративных информационных системах, где традиционные детерминированные методы недостаточно эффективны.

Литература

1. Fellegi I.P., Sunter A.B. A Theory for Record Linkage // Journal of the American Statistical Association. 1969. Vol. 64, No. 328. P. 1183–1210.
2. Winkler W.E. Overview of Record Linkage and Current Research Directions // Statistical Research Division, U.S. Census Bureau. 2006.
3. Berners-Lee T., Hendler J., Lassila O. The Semantic Web // Scientific American. 2001. Vol. 284, No. 5. P. 34–43.
4. Bizer C., Heath T., Berners-Lee T. Linked Data – The Story So Far // International Journal on Semantic Web and Information Systems. 2009. Vol. 5, No. 3. P. 1–22.
5. OASIS Universal Business Language (UBL) // URL: https://www.oasis-open.org/committees/ubl/ (дата обращения: 26.03.2026).
6. Ярмолик В.Н., Шевченко Н.А., Петровская В.В. Мера отличия для управляемых вероятностных тестов // Доклады БГУИР. 2022. Т. 20, № 6. С. 52–60.
7. Kuznetsov M.A., Nguen T.T. Mathematical Models of Web Resources Search // Prikaspiyskiy Zhurnal: Upravlenie i Vysokie Tekhnologii. 2013. Vol. 22. P. 25–30.
8. RapidFuzz // GitHub URL: https://github.com/rapidfuzz/rapidfuzz (дата обращения: 20.02.2026).
9. Вахлаков Д.В., Мельников С.Ю., Пересыпкин В.А. Многоэтапный метод автоматической коррекции искажённых текстов // Известия Южного федерального университета. Технические науки. 2020. № 7. С. 35–44.
10. Pandas // URL: https://pandas.pydata.org/ (дата обращения: 20.02.2026).

References in Cyrillics

1. Fellegi I.P., Sunter A.B. A Theory for Record Linkage // Journal of the American Statistical Association. 1969. Vol. 64, No. 328. P. 1183–1210.
2. Winkler W.E. Overview of Record Linkage and Current Research Directions // Statistical Research Division, U.S. Census Bureau. 2006.
3. Berners-Lee T., Hendler J., Lassila O. The Semantic Web // Scientific American. 2001. Vol. 284, No. 5. P. 34–43.
4. Bizer C., Heath T., Berners-Lee T. Linked Data – The Story So Far // International Journal on Semantic Web and Information Systems. 2009. Vol. 5, No. 3. P. 1–22.
5. OASIS Universal Business Language (UBL) // URL: https://www.oasis-open.org/committees/ubl/ (data obrashcheniya: 26.03.2026).
6. Yarmolik V.N., Shevchenko N.A., Petrovskaya V.V. Mera otlichiya dlya upravlyaemykh veroyatnostnykh testov // Doklady BGUIR. 2022. T. 20, № 6. S. 52–60.

7. Kuznetsov M.A., Nguen T.T. Mathematical Models of Web Resources Search // Prikaspiyskiy Zhurnal: Upravlenie i Vysokie Tekhnologii. 2013. Vol. 22. P. 25–30.
8. RapidFuzz // GitHub URL: https://github.com/rapidfuzz/rapidfuzz (data obrashcheniya: 20.02.2026).
9. Vakhlakov D.V., Mel'nikov S.Yu., Peresypkin V.A. Mnogoetapnyy metod avtomaticheskoy korrektsii iskazhennykh tekstov // Izvestiya Yuzhnogo federal'nogo universiteta. Tekhnicheskie nauki. 2020. № 7. S. 35–44.
10. Pandas // URL: https://pandas.pydata.org/ (data obrashcheniya: 20.02.2026).

Гильмутдинов Тимур Артурович, Магистрант, Московский физико-технический институт (национальный исследовательский университет), timurgilmutpr@gmail.com
Подлесных Дмитрий Артурович, Асппирант, Московский физико-технический институт (национальный исследовательский университет), ORCID 0000-0001-5992-0248 , Долгопрудный, Россия, podlesnykh.da@mipt.ru

Ключевые слова

вероятностный поиск, нечёткое сравнение строк, расстояние Левенштейна, разрешение сущностей, связывание записей, байесовская оценка, rapidfuzz, pandas, очистка данных, интеграция данных.

Gilmutdinov Timur Arturovich. Development of a Hybrid Method for Probabilistic Search and Verification of Records in Databases

Podlesnykh Dmitry Arturovich. Development of a Hybrid Method for Probabilistic Search and Verification of Records in Databases

Keywords

probabilistic search, fuzzy string matching, Levenshtein distance, entity resolution, record linkage, Bayesian estimation, rapidfuzz, pandas, data cleaning, data integration.

Abstract

The purpose of the research is to develop a hybrid method of probabilistic search and verification of records in relational databases, which allows matching string attributes containing typos, abbreviations and discrepancies. The methodology is based on calculating the normalized Levenshtein distance for each attribute, forming a Bayesian estimate of the a posteriori probability of matching a record from an external source with a reference record, and selecting a candidate who maximizes this probability. To improve performance, C extensions (rapidfuzz library), limiting the number of candidates to the top 20 most similar entries by full name, vectorization of operations using pandas, and pre-normalization of strings are used. Key results: the developed method demonstrates resistance to input data distortion, takes into account several attributes (full names, identifiers, names) at the same time and does not require a marked-up training sample, unlike the classical probabilistic Fellegi–Sunter approach. An experimental check on a reference base of 250,000 records and a test sample of 159 distorted records showed a processing time of 62 seconds on standard equipment. Conclusion: the proposed hybrid method combines the flexibility of fuzzy string comparison and probabilistic estimation, which makes it effective for data integration and cleaning tasks in corporate information systems.