University of Pardubice
Faculty of Electrical Engineering and Informatics
Department of Process Control

Similarity Space and Its Applications

Doctoral Thesis

Ing. Bc. Ondřej Rozinek

Ph.D. Programme: Electrical Engineering and Informatics
Supervisor: doc. Ing. Jan Mareš, Ph.D.
Co-supervisor: Mgr. Ing. Pavel Kříž, Ph.D.

Pardubice, April 2024

Thesis Supervisor:
doc. Ing. Jan Mareš, Ph.D.
Department of Process Control
Faculty of Electrical Engineering and Informatics
University of Pardubice
Studentská 95
532 10 Pardubice
Czech Republic

Thesis Co-supervisor:
Mgr. Ing. Pavel Kříž, Ph.D.
Department of Mathematics, Informatics and Cybernetics
Faculty of Chemical Engineering
University of Chemistry and Technology Prague
Technická 5
166 28 Prague 6–Dejvice
Czech Republic

Declaration

I declare that the thesis entitled Similarity Space and Its Applications is my own work. All literary sources and information that I used in the thesis are referenced in the bibliography.

I have been acquainted with the fact that my work is subject to the rights and obligations arising from Act No. 121/2000 Sb., On Copyright, on Rights Related to Copyright and on Amendments to Certain Acts (Copyright Act), as amended, especially with the fact that the University of Pardubice has the right to conclude a license agreement for the use of this thesis as a school work under Section 60, Subsection 1 of the Copyright Act, and that if this thesis is used by me or a license to use it is granted to another entity, the University of Pardubice is entitled to request a reasonable fee from me to cover the costs incurred for the creation of the work, depending on the circumstances up to their actual amount.

I acknowledge that in accordance with Section 47b of Act No. 111/1998 Sb., On Higher Education Institutions and on Amendments to Other Acts (Higher Education Act), as amended, and the Directive of the University of Pardubice No. 7/2019 Rules for Submission, Publication and Layout of Theses, as amended, the thesis will be published through the Digital Library of the University of Pardubice.

In Pardubice, April 2024

.....
Ing. Bc. Ondřej Rozinek

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor, doc. Ing. Jan Mareš, Ph.D., for granting me the opportunity to pursue my own topic and for his invaluable guidance and the wealth of experience he has shared from the academic environment.

I am also deeply grateful to the University of Pardubice, specifically the Faculty of Electrical Engineering and Informatics, for their support throughout my studies and for providing the necessary flexibility in my individual study plan.

I would like to extend my thanks to Mgr. Ing. Pavel Kříž, Ph.D., and Mgr. Šimon Axmann, Ph.D., for their assistance with the technical aspects of mathematical language and their valuable suggestions.

Lastly, my gratitude goes to the software technology company, Rozinet s.r.o. Without their support—comprising technical resources, financial assistance, data provision, and research space—this dissertation would not have been feasible.

Annotation

Mathematical spaces have been studied for centuries and belong to the basic mathematical theories, which are used in various real-world applications. In general, a mathematical space is a set of mathematical objects with an associated structure. This structure can be specified by a number of operations on the objects of the set. These operations must satisfy certain axioms of mathematical space.

Similarity and dissimilarity functions are widely used in many research areas: in information retrieval, data mining, machine learning, cluster analysis and applications in database search, protein sequence comparison and many more. When a dissimilarity function is used, a distance metric is normally required. On the other hand, although similarity functions are used, there is no formally accepted definition of this concept. In this dissertation is used for the first time the novel term similarity space.

A significant contribution of this dissertation is the identification of a class of functions that satisfy the axioms of similarity space, alongside the development of novel mathematical theorems and definitions that extend our understanding of similarity. This includes the exploration of duality between similarity and metric spaces, the introduction of normalization transformations that addresses to solution to open unsolved problem, and the establishment of new descriptions and definitions for convergence, continuity, and other fundamental properties within similarity spaces. A significant section is dedicated to developing a new fixed-point theory in similarity space, establishing solutions for differential equations, and introducing a new convergence criterion for the Newton method. Another theoretical contribution is the novel application of similarity space in linear regression.

Within the framework of Natural Language Processing (NLP) and Artificial Intelligence (AI), this dissertation applies theoretical insights to address real-world challenges, particularly in the areas of approximate string matching, complex fuzzy record matching and deduplication. By developing a novel convolution-based string matching model, proposing an advanced mathematical model for fuzzy record similarity, and introducing an optimal Q-gram filter for bipartite matching, this research presents novel solutions that significantly improve upon the state-of-the-art methods in terms of efficiency, accuracy, and applicability.

In conclusion, this dissertation not only advances the theoretical understanding of similarity spaces but also demonstrates their vast potential for application in data processing and analysis. By bridging the gap between abstract mathematical theory and practical computational challenges, this work lays the groundwork for future innovations across broad range of fields.

Keywords: similarity metric; similarity space; normalized similarity; edit distance; Jaccard coefficient; Q-gram filter; indexing method; approximate string matching; record linkage; entity resolution; record deduplication; similarity search; similarity join; linear regression; fixed point.

Anotace

Matematické prostory jsou studovány po staletí a patří k základním matematickým teoriím, které jsou využívány v různých aplikacích v reálném světě. Obecně je matematický prostor množina matematických objektů s přidruženou strukturou. Tato struktura může být specifikována řadou operací nad objekty množiny. Tyto operace musí splňovat určité axiomy matematického prostoru.

Funkce podobnosti a nepodobnosti jsou široce využívány v mnoha výzkumných oblastech: při vyhledávání informací, dolování dat, strojovém učení, shlukové analýze a aplikacích v databázovém vyhledávání, porovnávání sekvencí proteinů a mnoha dalších. Při použití funkce nepodobnosti je obvykle vyžadována metrika vzdálenosti. Na druhou stranu, i když jsou používány funkce podobnosti, neexistuje formálně přijatá definice tohoto pojmu. V této disertační práci je poprvé použit nový termín prostor podobnosti.

Významným přínosem této disertační práce je identifikace třídy funkcí, které splňují axiomy prostoru podobnosti, spolu s vývojem nových matematických vět a definic, které rozšiřují naše porozumění podobnosti. To zahrnuje výzkum duality mezi prostory podobnosti a metrickými prostory, zavedení normalizačních transformací, které řeší otevřený nevyřešený problém a stanovení nových popisů a definic pro konvergenci, kontinuitu a další základní vlastnosti v prostoru podobnosti. Významná část je věnována vývoji nové teorie pevného bodu v prostoru podobnosti, stanovení řešení diferenciálních rovnic a zavedení nového kritéria konvergence pro Newtonovu metodu. Dalším teoretickým přínosem je nové použití prostoru podobnosti v lineární regresi.

V rámci zpracování přirozeného jazyka (NLP) a umělé inteligence (AI) tato disertační práce aplikuje teoretické poznatky na řešení reálných problémů, zejména v oblastech přibližného vyhledávání řetězců, komplexní fuzzy porovnávání záznamů a deduplikace. Vývojem nového modelu pro vyhledávání řetězců založeného na konvoluci, navržením pokročilého matematického modelu pro fuzzy podobnost záznamů a zavedením optimálního Q-gramového filtru pro bipartitní matching tato výzkumná práce představuje nová řešení, která výrazně zlepšují stávající metody z hlediska efektivity, přesnosti a použitelnosti.

Závěrem, tato disertační práce nejen posunuje teoretické pochopení prostorů podobnosti, ale také demonstruje jejich široký potenciál pro aplikaci v zpracování a analýze dat. Tato práce překlenuje propast mezi abstraktní matematickou teorií a praktickými výpočetními výzvami a pokládá základy pro budoucí inovace v širokém spektru oborů.

Klíčová slova: metrika podobnosti; prostor podobnosti; normalizovaná podobnost; editační vzdálenost; Jaccardův koeficient; Q-gramový filtr; metoda indexace; přibližné vyhledávání řetězců; propojení záznamů; rozlišení entit; deduplikace záznamů; vyhledávání podobnosti; podobnostní spojení; lineární regrese; pevný bod.

List of Abbreviations

AI Artificial Intelligence

ANN All Nearest neighbors

BLAST Basic Local Alignment Search Tool

ConvJ Convolutional Jaro

ConvJW Convolutional Jaro-Winkler

DBSCAN Density-based spatial clustering of applications with noise

DNA Deoxyribonucleic acid

ECG Electrocardiogram

FASTA A data format and algorithms for fast biological sequence alignment and searching

FRS Fuzzy Record Similarity

LCS Longest Common Subsequence

MAE Mean Absolute Error

MSE Mean Squared Error

NLP Natural language processing

OLS Ordinary Least Squares

QSI Quantum Similarity Index

QSM Quantum Similarity Measure

XAI Explainable Artificial Intelligence

List of Symbols

$\mathcal{C}(A)$ Class of functions satisfying the axioms of similarity space

A_w Sum of weights for misalignments in convolutional similarity measures

$F_{\mathcal{R}}$ Discrete distribution functions $F_{\mathcal{R}_1}$ and $F_{\mathcal{R}_2}$ represent the ascending sorted lengths of tokens for sets of records $\mathcal{R}_1$ and $\mathcal{R}_2$, respectively

M_w Sum of weights for matches in convolutional similarity measures

P Probability measure

T Contraction mapping

$[\cdot]$ Denotes the s-norm of an element in a s-normed vector space

Δ Represents the difference operator

Ω Sample space

Σ^* The set of all finite strings over finite alphabet Σ

Σ Finite alphabet

α Represents a threshold for classification in matching/no-match scenarios or a contraction constant, depending on the context

$\bar{\delta}_{ij}$ Inverse Kronecker delta, equals 0 if $i = j$, and 1 otherwise

β Decay rate in exponential weighting function

χ The characteristic function of a set, equaling 1 for elements in the set and 0 otherwise

δ_{ij} Kronecker delta, equals 1 if $i = j$, and 0 otherwise

$\langle \cdot, \cdot \rangle$ Denotes the inner product of two elements

$\mathbb{E}[t_{\mathcal{M}}]$ Expected minimum shared Q-grams in unsupervised learnable Q-gram filters

$\mathbb{N}$ The set of all natural numbers

$\mathbb{R}_{>0}$ The set of positive real numbers

$\mathbb{R}_{\geq 0}$ The set of non-negative real numbers

$\mathbb{R}$ The set of all real numbers

$\mathcal{F}$ σ-algebra of events

$\mathcal{H}$ Denotes a Hilbert space

$\mathcal{O}$ Big O describes maximum time complexity

$\mathcal{R}$ A generic record or a set representing an entity in the database. Specific instances of records are denoted by subscripts, e.g., $\mathcal{R}_1$, $\mathcal{R}_2$

$\mathcal{X}$ Non-empty set

∇ Denotes the gradient operator

σ Standard deviation parameter in Gaussian weighting function

$\widehat{t}_{\mathcal{M}}$ Approximate Q-gram count for matching records, using sorted lengths and fixed α

$\|\cdot\|$ Denotes the norm of an element in a normed vector space

s_{ConvJW} Convolutional Jaro Winkler similarity measure

s_{ConvJ} Convolutional Jaro similarity measure

s_{JW} Jaro-Winkler similarity measure

s_J Jaro similarity measure

t_α Minimum Q-grams to meet similarity threshold α between strings X and Y, considering edit distance and string lengths

$t_{\mathcal{M}}$ Minimum shared Q-grams between records $\mathcal{R}_1$ and $\mathcal{R}_2$ for threshold α, considering bipartite matching and token size differences

$\mathcal{E}$ Entity collection

$\mathcal{F}_\alpha$ Filtering function

$\mathcal{M}_\alpha$ Matching function

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

This doctoral thesis (hereinafter referred to as "the thesis") is concerned with similarity space and its applications. The motivation for the thesis is first explained. Since 2010, research has been conducted by the author on improving database search algorithms and enhancing fuzzy matching algorithms for so-called dirty data [1] collected from various sources. Through work with users and the knowledge gained from studying state-of-the-art issues, the author has gradually gained insight and connections between theory and practice over the years of this research. In the last years of research in designing increasingly optimal algorithms, a very simple but fundamental question was first posed by the author, which initiated this work. This question focused on the search for the optimality of similarity between a query in a search engine and its answer, particularly from the perspective of dirty data, necessitating primarily the handling of textual similarity between database records. The question was:

"What is an ideal similarity function, and what are the properties of such an ideal function?"

In the extensive effort and search for a description of such an ideal function from a mathematical point of view and their properties, the author has extensively explored mathematical spaces and functions that could model this for modern and increasingly desired user cases. A question posed in this manner may seem very vague, and some might argue that such an ideal function does not exist or is unattainable. However, humanity might harbor some ideals and at least attempt to approach them. As evidenced in the research below, it has been concluded by the author that there are certain common ideal properties for a large class of such functions, and it is believed that some part of such a goal can be achieved.

Mathematical spaces have been studied for centuries and belong to the basic mathematical theories, which are used in various real-world applications [2]. Generally, a mathematical space is defined as a set of mathematical objects with an associated struc-

ture. This structure can be specified by a number of operations on the objects of the set. These operations must satisfy certain axioms of mathematical space. The mathematical construction of metric space and similarity space are based on topological space, which, in turn, is founded on set theory [3]. Nowadays, research groups all over the world are engaged in dealing with similarity spaces in various research fields, e.g., [2], [4]–[6].

For readability and to reach a broad audience, not all mathematical circumstances and conditions are treated in detail. Instead, the main concept and a pathway to a solution are presented by the author. Grasping all the current theories would take an excessive amount of time, consequently leaving no time for innovation. This constraint highlights the necessity for the author to prioritize areas of research that promise the most significant contributions to knowledge and practical applications. Due to the limited scope of this thesis, it is not feasible to cover all related fields comprehensively. These contributions are expected to undergo further refinement and enhancement. The author refers readers to foundational texts for the fundamental concepts and properties of topological and metric spaces, such as convergence, continuity, completeness, separability, connectedness, compactness, etc. [5], [7]–[11].

Similarity and dissimilarity functions are widely utilized in numerous research areas, including information retrieval, data mining, machine learning, cluster analysis, and applications in database search, protein sequence comparison, and more. While a distance metric is normally required when a dissimilarity function is used, no formally accepted definition exists for similarity functions, despite their usage [2], [4], [6].

1.1 Why Similarity Space?

Similarity theories are extensively applied in a wide array of human endeavors and scientific fields, including but not limited to nature, physics, chemistry, biology, earth science, environmental science, engineering, geology, psychology, sociology, astronomy, art, economics and computer science. Traditionally, these applications have relied on a diverse set of indices, coefficients, measures, and dissimilarity functions, which often utilize a non-intuitive and indirect notion of distance [5]. This thesis presents a new, universally applicable theory of similarity space [4], [6], [12], [13], [OR-1], developed to bridge and synthesize the diverse fields mentioned, whether considered separately or together.

Key motivations for developing the theory of similarity space include:

- **Unified Framework** Currently, diverse fields rely on a patchwork of measures (indices, coefficients, etc.) for similarity, often lacking a rigorous mathematical foundation. A theory of similarity space would provide a unified framework for measuring

similarity across all disciplines, promoting consistency and communication [5], [14], [15].

Example (Unified Climate Analysis)**.** *In climate modeling within environmental science, disparate indices assess variables like temperature anomalies and vegetation changes, complicating integrated analysis. A similarity space framework could unify these measures, enabling comprehensive climate change assessments by providing a standardized approach to compare different environmental variables. This would facilitate better synthesis of data, improving understanding of regional climate impacts and informing conservation strategies through a common similarity [16]–[18].*

- **Directly Addressing Similarity and Self-Similarity** Existing approaches often rely on indirect notions of distance (dissimilarity). A formal theory could explicitly model both similarity between objects and self-similarity within objects, leading to a more intuitive and interpretable understanding [4]–[6], [12], [13], [OR-1], [19]–[21].

  Example (Quantifying Linguistic Evolution)**.** *Linguistic studies often compare languages to understand their evolution, traditionally analyzing phonetic and grammatical features. This indirect method complicates the identification of linguistic relationships. Employing a similarity space theory could directly quantify inter-language similarities and intra-language self-similarities, offering a precise, quantitative basis for classifying languages. This approach would streamline linguistic research, enhancing the accuracy of language family classifications and uncovering new insights into linguistic diversity and evolution [22]–[24].*

- **Precise and Well-Defined Similarity:** Current methodologies often lack clear mathematical properties and expected behavior. A theory of similarity space would aim to develop more precise measures with well-defined properties, ensuring consistent and reliable results [4]–[6], [OR-1], [21], [25]–[27].

  Example (Enhancing Molecular Similarity Measures in Drug Discovery and Material Science)**.** *In computational chemistry, the similarity between molecular structures is crucial for drug discovery and material science. Traditional similarity measures, such as Tanimoto coefficients based on chemical fingerprints, offer a basic approach but may not fully capture the physicochemical properties that determine molecular behavior in biological systems. A theory of similarity space could refine this by introducing precise, mathematically grounded measures that consider a broader range of molecular features, including electronic properties, spatial orientation, and reactive potential. This enhanced precision would enable more reliable*

predictions of molecular interactions, significantly advancing the fields of pharmacology and materials engineering [25]–[27].

- **Increasing Scalability and Efficiency:** Current similarity measures often struggle with large or complex datasets. A theory of similarity space could lead to the development of more scalable and efficient methods for measuring similarity. This would be particularly beneficial in fields dealing with massive datasets, such as astronomy, bioinformatics, or social media analysis [28]–[30].

  Example (Scaling Similarity Analysis)**.** *Imagine analyzing the similarities between millions of galaxies, protein structures, or user profiles. A theory of similarity space could provide a framework for developing efficient algorithms that can handle such large-scale comparisons effectively. This would not only save time and resources but also allow researchers to extract valuable insights from these vast datasets [31].*

- **Consolidating Current Approaches** By offering a general framework, the theory could consolidate current methodologies, reducing complexity and potentially leading to new discoveries through the unification of diverse approaches [32]–[35].

  Example (Bridging the Gap Between Symbolic and Subsymbolic Representations)**.** *Many fields deal with data that can be represented in two ways: symbolically (explicit labels or categories) and subsymbolically (raw data without explicit meaning). A theory of similarity space could provide a framework for comparing and relating these different representations. This would be valuable in areas like machine learning, cognitive science, and natural language processing [32], [33], [35].*

- **Enabling New Applications and Discoveries:** The current patchwork of similarity measures often limits the ability to compare data across disciplines. A unified theory could unlock new applications by allowing researchers to compare and analyze data from different fields in a more meaningful way. This cross-pollination of knowledge could lead to unexpected discoveries and breakthroughs [36]–[39].

  Example (Cross-Disciplinary Insights Through Unified Theory)**.** *For example, with a unified theory, researchers in biology might be able to leverage insights from physics or engineering to understand complex biological systems. Similarly, economists might be able to utilize concepts from sociology to develop more precise models of human behavior [36]–[39].*

- **Formalizing Subjectivity in Similarity Judgments:** Many applications rely on human judgments of similarity, which can be subjective and context-dependent.

A theory of similarity space could provide a framework for incorporating and formalizing this subjectivity. This would be valuable in fields like art history, music theory, or product design, where aesthetic similarity plays a crucial role [40]–[43].

Example (Modeling Human Perception)**.** *Current methods often struggle to capture the nuances of human perception. A theory of similarity space could allow researchers to develop models that account for factors like cultural background, personal preferences, or the specific context in which an object is being evaluated. This would lead to a more nuanced understanding of how humans perceive similarity and could be used to develop better tools for applications that rely on subjective judgments [40]–[43].*

- **Improved Explainability and Transparency:** Existing similarity measures often lack clear interpretations. A theory of similarity space could lead to more interpretable measures, making it easier to understand why two objects are considered similar. This transparency would be crucial in various fields, especially for applications with ethical considerations or where human oversight is essential [44]–[47].

  Example (Enhancing Explainability in AI)**.** *As AI systems become increasingly complex, ensuring their decisions are explainable and interpretable is crucial. Current similarity measures often lack transparency, making it difficult to understand how AI systems arrive at their conclusions. A theory of similarity space could be a framework for the development of more interpretable similarity measures. For instance, in image analysis, a unified framework for representing image similarity could reveal how AI models prioritize specific features like color, texture, or shapes when classifying images. This would significantly benefit research in Explainable Artificial Intelligence (XAI) [44]–[47].*

While fractal theory and various dissimilarity functions exist, they often approach the concept of similarity in a non-intuitive and indirect manner, primarily through the notion of distance. This thesis marks the first attempt to clearly define the concept of similarity space and explores its potential applications within the context of other theories and practical applications, specifically in computer science.

The development of the theory of similarity space facilitates the modeling of similarities across a broad spectrum of human activities and scientific disciplines (see Figure 1.1), offering a new perspective and tools for understanding and analyzing the interconnectedness of diverse phenomena.

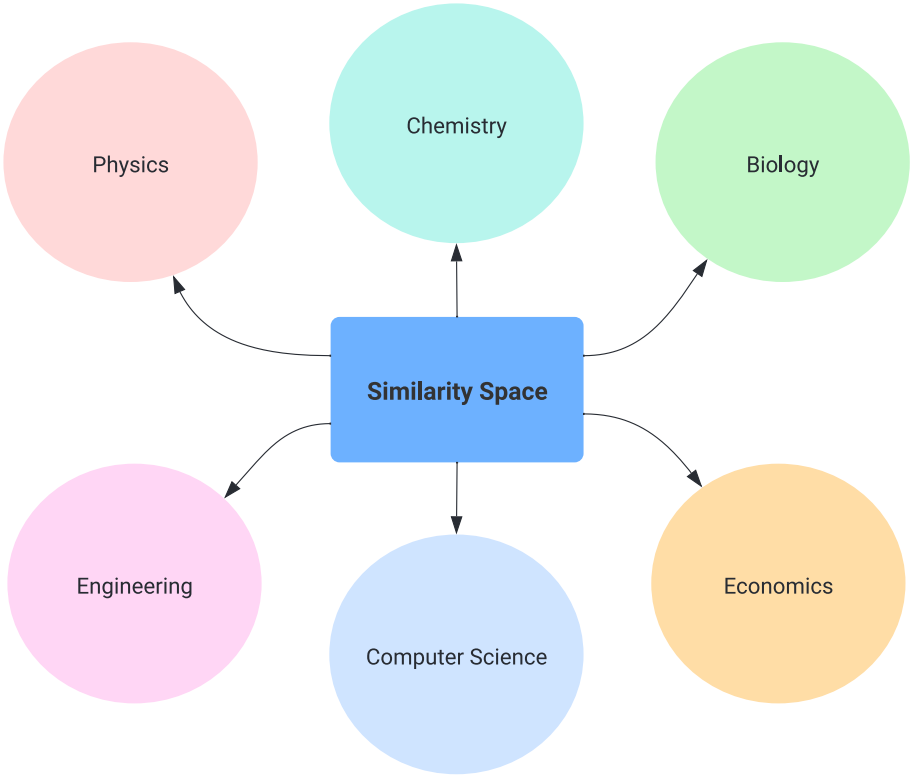

Figure 1.1: Applications of Similarity Space in Various Human Activity Fields and Beyond

Similarity in Nature Fractals [20] are intricate structures that repeat themselves at different scales, and they are surprisingly common in nature. These self-similar patterns can be found in various forms and scales, from tiny molecules to large ecosystems. Here are some examples of where fractals can be observed in nature:

- *Plants and Trees:* The branching patterns of trees and plants often exhibit fractal-like structures. For example, a single tree branch looks similar to the whole tree, and this pattern repeats at various scales from the trunk to the smallest twig [48].
- *Ferns:* Fern leaves are classic examples of natural fractals. Each fern leaf (frond) is made up of smaller leaves (pinnae), which in turn are made up of even smaller leaves, and this pattern continues down to very tiny scales [49].
- *Romanesco Broccoli:* This vegetable is famous for its natural approximation of a fractal pattern. Each bud is composed of a series of smaller buds, all arranged in a logarithmic spiral. This self-similar pattern continues at several levels of scale [49].
- *Coastlines:* The concept of fractal geometry was famously applied by Benoit Mandelbrot to describe the complex, self-similar patterns of coastlines, which look similar at different scales. The length of a coastline can seem almost infinite, as the level of detail increases with closer observation [19].

- *Mountains and River Networks:* The structure of mountain ranges and the branching patterns of rivers and their tributaries exhibit fractal-like properties. These patterns can be observed from high above the earth, in maps, or satellite images [20], [50].

- *Clouds and Atmospheric Phenomena:* The formation of clouds often displays fractal behavior, with cloud formations showing self-similarity across different scales. Lightning bolts also exhibit fractal patterns as they branch through the sky [51].

- *Snowflakes:* Each snowflake has a unique, six-fold symmetry with intricate patterns that can exhibit fractal-like structures, especially as the crystals branch out in repeating patterns [52].

- *Animal Coloration and Patterns:* The patterns on certain animals, such as the spots on a leopard or the stripes on a zebra, can exhibit fractal-like patterns. These patterns can be used for camouflage or for signaling to other animals [53].

- *Blood Vessels, Nervous Systems, and ECG Patterns:* The branching patterns of blood vessels and nerves in animals, including humans, show self-similar fractal patterns. This efficient branching helps maximize the reach of the circulatory and nervous systems within the body [54]. Similarly, the electrocardiogram (ECG) reveals a recurring self-similar pattern in the PQRST complex, indicative of the heart's electrical activity [OR-2].

- *Galaxies:* On a much larger scale, the distribution of galaxies in the universe can exhibit fractal-like patterns, with galaxy clusters forming filaments and voids that resemble fractal structures in nature [55].

  Example (Galaxies as Similarity Spaces)**.** *Consider visualizing a galaxy as a space of similarity in Figure 1.2, with its center representing the point of maximum mass concentration, analogous to the peak of similarity. This concept captures mass concentration more accurately than traditional metric space, which is typically centered at zero. Abstractly, this approach provides a better understanding of galaxies as entities where mass and similarity converge.*

Figure 1.2: Depiction of a galaxy, image generated by AI(DALL·E by OpenAI).

Fractals in nature are not only visually fascinating but also serve functional purposes, such as maximizing surface area for heat exchange in lungs or optimizing light absorption in leaves. The study of natural fractals bridges mathematics, physics, and biology, offering insights into the underlying principles that govern natural patterns and structures.

Similarity in Physics Physics employs the concept of similarity in several contexts, including similarity in fluid dynamics and thermodynamics. The principle of similarity allows physicists to predict the behavior of physical systems under varying conditions by using scaled models or simulations. For example, the Reynolds number is a dimensionless quantity in fluid mechanics that describes the similarity between the flow of fluids in different systems, helping to predict flow patterns in both laminar and turbulent systems [56], [57]. Interestingly, concept is notably linked to the challenging and unresolved Navier-Stokes problem, a Millennium Prize Problem as designated by the Clay Mathematics Institute, indicating a gap in current theories to adequately explain and model certain phenomena [58]. Additionally, the areas of stochastic heat equations [59], [60] and fractional Brownian motion [61]–[64] offer valuable opportunities for the application of similarity theories and the possibility for substantial progress in forthcoming research. The field of quantum mechanics is another domain where similarity theories find significant application, particularly in the study of subatomic particles. Through the use of quantum similarity measures, scientists can compare the electron densities of different molecules. This comparison is crucial for understanding the reactivity and properties of molecules, as electron density distributions play a vital role in determining molecular

behavior during chemical reactions. By applying similarity principles, researchers can predict how molecules will interact based on the similarities in their electron configurations, facilitating advancements in chemistry and materials science [65]. For instance, they refer to it as the Quantum Similarity Measure (QSM) and Quantum Similarity Index (QSI), reflecting the general nature of these terms ('index' or 'measure') in the absence of a specific theory of similarity to support their precise definition [66].

Similarity in Chemistry In chemistry, similarity can refer to the likeness in the structural or functional aspects of molecules. Molecular similarity is crucial in the field of drug discovery and development, where researchers look for compounds with structural similarities to known active molecules to find new therapeutic agents. For instance, molecular structures are represented as binary vectors (fingerprints), where each bit represents the presence or absence of certain substructural features. The similarity between molecules can be quantified using methods such as the Tanimoto coefficient, Jaccard index, Cosine similarity, Russell-RAO coefficient, Forbes coefficient, Soergel coefficient or Hamming distance [26], [67], [68]. This thesis will demonstrate that some of these coefficients and indices occupy the same similarity space, exhibit similar properties, and can be encompassed and replaced by a generalized form of similarity known as the *Generalized Rozinek similarity* [OR-1].

Similarity in Biology Biology employs the concept of similarity at various levels, from the macroscopic to the microscopic. At the macroscopic level, similarity in morphological traits can indicate common ancestry or evolutionary relationships among organisms. At the genetic level, similarity in DNA sequences is utilized to infer genetic relationships, evolutionary histories, and the functions of genes and proteins. Methods like Needleman-Wunsch [69] and Smith-Waterman [70], based on dynamic programming, demonstrate the utility of similarity properties in understanding biological sequences. Additionally, the concept of similarity extends to the study of protein structures through techniques such as BLAST [71] and FASTA [72], which enable the rapid identification of structural and functional similarities among proteins. Furthermore, the application of similarity concepts in phylogenetics, through methods like Maximum Likelihood [73] and Bayesian Inference [74], has revolutionized understanding of evolutionary relationships. These approaches leverage similarity in genetic sequences to construct phylogenetic trees, elucidating the evolutionary pathways of various species. In this thesis, the novel explanation is provided on how fundamental dynamic programming algorithms (Levenshtein, LCS) lie within a similarity space and can be generalized under *Generalized Rozinek similarity*.

Similarity in Earth Science In earth sciences, particularly within meteorology and climatology, the concept of similarity is utilized to analyze weather patterns and contribute to the development of climate models. Similarity theories facilitate the analysis of momentum, heat, and mass exchanges between the Earth's surface and the atmosphere, which are critical for accurate weather forecasts and climate change simulations. This concept is also applied in geomorphology to study the processes of landscape formation and erosion. For example, the Monin-Obukhov similarity theory [75] offers a theoretical basis for the investigation of atmospheric turbulence through similarity criteria. In the field of hydrology, hydraulic similarity principles are employed in the scaled modeling of river and estuary water flows, supporting flood management strategies and the construction of hydraulic infrastructures [76]. Furthermore, the study of plate tectonics and seismic dynamics benefits from similarity-based modeling techniques, which help in forecasting seismic events [77]. The application of similarity theories in these varied sub-disciplines of earth sciences enhances the comprehension of complex environmental and geological systems.

Similarity in Environmental Science In environmental science, the application of similarity concepts is essential for understanding ecosystems, biodiversity, and the assessment of environmental changes. Similarity indices [78], including the Sørensen-Dice index, Ochiai index, Anderberg index, Kulczynski index, Kulczynski-Cody index, Lennon index, and the Bray-Curtis dissimilarity [79], quantitatively evaluate the resemblance between ecological communities. This evaluation is critical for assessing species diversity and compositional similarities, enabling researchers to identify areas of significant biodiversity and ecosystems requiring urgent conservation efforts. The utility of similarity indices extends beyond terrestrial ecosystems, encompassing aquatic and marine environments where they are critical for evaluating the health of coral reefs, assessing the biodiversity impacts of pollution, and analyzing freshwater ecosystem dynamics. For instance, similarity measures are employed to understand the intricate balance of marine biodiversity and the effects of anthropogenic stressors on aquatic life forms [80], [81]. In addition to biodiversity assessment, similarity concepts are integral to landscape ecology, where they inform the study of habitat connectivity, fragmentation, and the spatial distribution of ecosystems. These analyses are essential for developing conservation plans that mitigate habitat loss and promote the preservation of ecological corridors vital for species migration and genetic diversity [82], [83]. Furthermore, similarity-based approaches are instrumental in climate change research, facilitating the comparison of past and present ecological conditions to predict future environmental scenarios. This application is crucial for understanding how changing climate variables influence biodiversity patterns and

ecosystem resilience, guiding adaptive management and conservation policies [84].

Similarity in Engineering In engineering, the principle of similarity is fundamental to the design and testing phases of product development. Utilizing scale models and established similarity laws, engineers are equipped to forecast the behavior of large-scale structures, vehicles, and systems based on the observations made from smaller, more manageable prototypes. This methodology is exceptionally beneficial in fields such as aerospace and civil engineering, where conducting tests on full-scale models can be impractical or prohibitively expensive. For instance, in aerospace engineering, wind tunnel testing of scale models based on Reynolds number similarity allows for the analysis of aerodynamic properties without the need for full-sized aircraft tests [85]. Similarly, in civil engineering, hydraulic modeling uses Froude number similarity to predict the flow of water around structures in projects such as dams and bridges [86]. Additionally, in the automotive industry, scale model crash testing employs similarity principles to enhance vehicle safety while minimizing development costs [87].

Similarity in Computer Science In computer science, similarity measures are crucial in various algorithms and applications, including machine learning, data mining, and information retrieval.

Similarity measures help in clustering and classification tasks, where the goal is to group together data points based on their likeness. One notable area of application is in natural language processing (NLP), where similarity measures are employed to evaluate the proximity between textual documents or words, facilitating tasks such as document classification, sentiment analysis, and machine translation [88], [89].

In machine learning, algorithms like k-nearest neighbors (k-NN) [90] rely on similarity measures to classify data points based on the likeness to their nearest neighbors in the feature space. Cosine similarity and Jaccard index are among the most commonly used metrics for assessing the similarity between vectors, which can represent text documents in NLP or user profiles in recommendation systems.

Application of Similarity Space in NLP In this thesis, it will be demonstrate the applications of similarity space theory only in narrow field in the context of approximate string matching. fuzzy record matching and deduplication, focusing on their utility for error-tolerant real-time search and deduplication tasks.

The exploration of similarity space theory provides a foundational framework for understanding how data can be compared and matched with a degree of tolerance for errors, which is crucial in handling real-world data that may contain inconsistencies, typos, or

variations in formatting. This thesis applies similarity space theory to fuzzy record matching and deduplication, aiming to present effective methodologies for identifying and linking records that correspond to the same entity across various databases, notwithstanding data discrepancies. Furthermore, the application of similarity space theory to approximate string matching elucidates techniques for searching and retrieving information based on strings that match approximately, thus improving the efficiency and accuracy of search functionalities within databases and information systems. These applications highlight the significance of similarity measures in enhancing data quality and accessibility, especially in tasks that demand high precision and speed.

1.2 Structure of the Thesis

The organization of this thesis is structured as follows.

The chapter 1 sets the stage by outlining the thesis's goals, introducing the concept of similarity space, and summarizing the major contributions made throughout the thesis. This chapter focuses primarily on problem formulation and the state-of-the-art in this area. The basic objectives of the dissertation are presented and structured into theoretical and practical parts in section 1.3. The subsequent chapters are divided into topics within research fields under the overarching theme of similarity space, highlighting the author's significant contributions.

The chapter 2 develops the concept of similarity space. The section 2.1 presents a comprehensive review of the current state of similarity space research (subsection 2.1.4), introducing the novel concept of similarity space (section 2.2) and discussing its foundational theory. This chapter also explores the duality between similarity and metric spaces (section 2.5).

The chapter 3 introduces a novel perspective on linear regression within similarity space, offering fresh insights and methodologies.

In chapter 4, the novel fixed point theory in similarity space is presented, including a description of the current state of the art in related metric spaces and showcasing some applications of this theory in similarity space.

The chapter 5 focuses on a specific application of similarity space for approximate string matching, record matching, and record deduplication, demonstrating the practical utility of similarity space concepts.

The chapter 6 synthesizes the achievements of the thesis, discussing the significance and impact of the findings within the broader scientific community.

The chapter 7 summarizes the key findings, contributions, and suggests directions for future research.

1.3 Aims of the Thesis

The objectives of this thesis are divided into theoretical and practical parts. In the theoretical part, the focus is on the unification and further development of the theory of similarity space. The practical part aims to apply this theory in the fields of approximate string matching, approximate record matching and deduplication within similarity space.

The classical theories of metric space and partial metrics have been extensively studied. However, the behavior within similarity space remains largely unexplored and is believed to possess significant potential for broad applicability.

Main Goals

The main goals of this dissertation are to unify and further develop the theory of similarity space and apply this theory to practical fields such NLP (approximate string matching, record matching, and deduplication within similarity space).

In this thesis, the focus in the theoretical part will be mainly on:

- The study of the transformation between similarity space and metric space as a monotonically decreasing transformation, referred to as duality.
- The examination of normalization transformations that preserve similarity as an open unsolved problem [2], [6].
- The identification and proof of similarity functions belonging to a class of functions that satisfy the axioms of similarity space, denoted by $\mathcal{C}(\mathcal{A})$.
- The introduction of new descriptions and definitions of topology, convergence and continuity in similarity space.
- The new perspective on traditional methods, such as linear regression.
- The presentation of a new fixed point theory in similarity space, proving its existence, uniqueness, and convergence.

In the practical part, attention will be given to:

- The application of similarity space to approximate string matching.
- The development of an applicable model for complex fuzzy record matching on a bipartite graph.
- The creation of a new Q-gram filter as a lower bound for complex record fuzzy matching based on bipartite graph.
- The development of a fuzzy deduplication model for scalable solutions in Big Data.

1.4 Contributions of the Thesis

In this section, the major contributions of the author's dissertation are listed in sequential order by chapter. While some minor contributions are not explicitly named, they can be found throughout the thesis and the author's articles.

1.4.1 Similarity Space Theory

- **Formal Definition of Similarity Space**
  - Introduction of new defined term *similarity space*, based on research similarity within a broad mathematical context, elevating it to the level of metric spaces and other significant spaces in mathematics.
  - Discussion and establishment of axiomatic systems of similarity space (Definition 5).
  - Proposal of a specific axiomatic system for normalized similarity space (Definition 6) [OR-1].
- **Duality in Similarity and Metric Space**
  - New duality Theorem 10 proposed between similarity and metric spaces, presenting a viable theory on possible transformations between these spaces and addressing how to achieve this [OR-1] .
- **Discovering Some Theorems**
  - Linear Transformation (Theorem 3) - Proof that positive linear transformations of a similarity result in a new similarity [OR-1].
  - Convex Combination (Theorem 5) - Proof that convex combination of normalized similarity is itself a normalized similarity [OR-1].
- **Embeddings into Similarity Space**
  - Similarity of Two Objects (Theorem 12) - Establishing the intersection of two objects as a similarity [OR-1]. Discussion on transforming distance into similarity and vice versa, as elaborated in Corollary 4 and Corollary 2) [OR-1].
  - The formal framework for interpreting normalized similarity s_n as analogous to a probability measure enriches the understanding of similarity within the probability theory. The integration of similarity measures into probability spaces through the event intersection model provides a solid foundation for analyzing and applying similarity in a probabilistic context (Theorem 13).

  - The construction of a similarity space within a Hilbert space framework, utilizing an inner product to define a similarity, demonstrates a novel approach to measuring similarity in vector spaces (Theorem 14).

- **Introduction of S-Norm in Functional Analysis**

  - The innovation of the s-norm (Definition 18) within functional analysis signifies a contribution extension of norm theory, offering a novel measure for evaluating magnitude through the infimum. This new functional analysis tool not only enriches the mathematical landscape of norms but also introduce alternative ways to explore function space and solving differential equations.

- **Class Functions Belonging to Similarity Space**

  - It is important to note that the Jaccard index, due to its absence of a defined space, is accordingly referred to as an index. The current state-of-the-art requires its conversion into a distance measure, as shown in Kosub's article [91]. This thesis demonstrates its direct assignment to the similarity space without needing any transformation.
  - Development of a generic similarity equation named *Generalized Rozinek Similarity* (Theorem 18).
  - Proofs that the Tanimoto Coefficient (Theorem 21), Gaussian Similarity (Theorem 24), Ruzicka Similarity (Theorem 23), Normalized Edit Similarity (Theorem 45), and Longest Common Subsequence (LCS) belong to this class of function [OR-1].

- **Solution to Unsolved Open Problem**

  - Chen [6] and Elzinga [2] identified the general solution for constructing a similarity from a distance as an *unsolved open problem*. The Generalized Rozinek Similarity [OR-1] is presented as a generic solutions with proven direct relation to Jaccard index, Tanimoto coefficient, Ruzicka similarity, normalized edit similarity and LCS.

1.4.2 Linear Regression in Similarity Space

- **Objective Function in Similarity Space**

  - Introduced an innovative method to adapt traditional least squares regression for use in similarity spaces in Definition 21, focusing on textual data analysis.

This adaptation is critical for analyzing qualitative aspects of data, such as word similarities in NLP taks [OR-3], [OR-4].

- **Proposed Optimization Algorithm**
  - Proposed a novel optimization algorithm, outlined in Algorithm 1, for linear regression models in similarity spaces. This algorithm utilizes dot product maximization with regularization to optimize model parameters, as shown in the equation 3.11.
  - These contributions significantly advance the field of data analysis by providing new tools and methodologies for understanding and interpreting complex patterns in textual data, bridging the gap between quantitative precision and qualitative depth [OR-3], [OR-4].

1.4.3 Fixed-Point Theory in Similarity Space

- **Topology of Similarity Space**
  - Introduces an elementary metric (Theorem 6) and a quasi-metric (Theorem 1) in a dualistic perspective within similarity spaces, complete with proofs.
  - Discusses the introduction of an induced elementary similarity (Theorem 2) and its relationship to intersections in a measure space context.
  - Introduces fundamental concepts such as open s-balls and closed s-balls, crucial for defining topology in similarity spaces (Definition 7).
  - Provides proof that open s-balls in similarity spaces form a topological basis (Theorem 7).
  - Proposes a new definition of topology in similarity space, termed similarizable space, and proves that any similarizable space is a Hausdorff space (Theorem 8).
- **Convergence and Continuity in Similarity Space**
  - Defines convergent sequences in similarity spaces (Definition 11).
  - Proves the duality of convergent sequences in dual similarity spaces (Lemma 4).
- **Introduction of New Definitions and Theorems**
  - Develops new concepts like s-continuity, s-derivative (Definition 14) extending traditional mathematical notions to similarity spaces.

  - Presents the Sequential Criterion for s-Continuity.

- **Existence, Convergence and Uniqueness in Contraction Principle**

  - Examines the conditions under which a function in a similarity space will have fixed points, extending classical concepts from functional analysis into similarity spaces.
  - Addresses the conditions for the existence and uniqueness of fixed points for mappings in similarity spaces, analogous to the Banach Contraction Principle (Theorem 39 and Theorem 23) with proofs.
  - Introduces new Boyd-Wong theorems in similarity space, such as the Boyd-Wong Dualistic Contraction (Theorem 41) and Boyd-Wong Similarity Contraction (Theorem 42), detailing bifurcated conditions for self-similarity and mutual similarity, along with proofs of existence, uniqueness, and Picard sequence convergence.

- **Applications in Various Fields**

  - Discusses the general applicability in space contraction, such as the existence of solutions in different types of differential equations.
  - Demonstrates the applicability of the theory in solving non-linear in-homogeneous Fredholm integral equation of the second kind (Theorem 43), and adapting iterative methods like Newton's method for similarity spaces (Example 4.75).

1.4.4 Entity Resolution in Similarity Space

- **Convolutional-Based String Matching**

  - Introduced general and very flexible convolutional-based string matching model to enhance unsupervised character-based approximate string matching accuracy and efficiency (Definition 25) [OR-5].
  - Generic modelling of positional character proximity and character weight importancy accross string in quasi-linear time complexity $\mathcal{O}(|S1|w)$ which has faster execution time than traditional methodology utilizing dynamic programming [OR-5].

- **Convolutional Jaro Similarity**

  - Developed ConvJ (Definition 27), incorporating convolutional methods with Gaussian weighting, detailed in Algorithm 4.

  - Achieved superior accuracy and maintained computational efficiency, with approximately a 10% margin improvement in F1-score compared to state-of-the-art unsupervised approximate string matching algorithms [OR-5].
  - Provided the fastest execution time compared to other state-of-the-art algorithms with the quasi-linear time complexity $\mathcal{O}(|S_1|w)$ [OR-5].

- **Convolutional Jaro-Winkler Similarity**
  - Introduced ConvJW (Definition 28), an advancement of Jaro-Winkler with convolutional techniques, as outlined in Algorithm 4.
  - Introduced exponential decay weighting for characters in the string to model characters as more important at the start of the string.
  - Achieved the best accuracy in experiments on real data, as shown in Table 5.3.4 [OR-5].

- **Fuzzy Record Similarity**
  - An advanced method known as the Fuzzy Record Similarity (FRS) is developed. This is the first advanced mathematical model of an approximate record similarity that fulfills the axioms (S1), (S2), (S3), and (S4) of a similarity space, as defined in Definition 5. With its robust mathematical properties, FRS is ideally suited for extensive use in text mining and cluster analysis. The model uniquely omits the tuning of the second threshold parameter δ, demonstrating impressive results on real datasets compared to baseline state-of-the-art methods [OR-6].

- **Optimal Q-gram Filter**
  - A mathematical model of an optimal Q-gram filter for bipartite matching on sets of tokens $|\mathcal{R}_1|$ and $|\mathcal{R}|_2$ is proposed. This optimal count Q-gram filter acts as the greatest lower bound for the number of shared Q-grams at threshold α. Recognized as an infimum for shared Q-gram count, it represents the most efficient filter achievable. Theorem 49 demonstrates that achieving this optimal filter corresponds to an integer linear programming problem, maximizing expected distances across tokens in the worst-case scenario. Integer linear programming tasks with related constraints for FRS, Fuzzy Cosine, Fuzzy Jaccard, and Fuzzy Dice are formulated [OR-6].

- **Approximate Q-gram Filter**

  - Given that calculating the optimal Q-gram filter results in a linear programming calculation, which is not easily analytically solvable, a substantial approximation is developed, as stated in Theorem 17. This approximation avoids the optimization problem and can be analytically solved through a few simplification steps. These findings show that this filter operates at a constant time complexity $\mathcal{O}(1)$. Its high accuracy and real-time capabilities were confirmed experimentally, demonstrating excellent precision in filtering, generating a small candidate set, and processing all datasets in 220 milliseconds. This establishes it as the most accurate analytically solvable Q-gram lower bound for bipartite matching that runs in constant time. Proposed Algorithm 8 of BipartiteJoin for two step probabilistic similarity join [OR-7].

- **Singularity and Efficiency of Q-gram Filter**

  - Interesting properties of the Q-gram filter are discussed, particularly its effectiveness related to string lengths and the selected threshold α. Formulas for the singularities of the filter and the minimum effective threshold α are derived, as exemplified in Theorem 51 [OR-6].

- **Padding Extension of Q-gram Filter**

  - The benefits of padding extension and its incorporation into the model are explored. This enhances the F-measure of Q-gram similarity by smoothing token boundaries, expanding the feature set, and increasing the minimum shared Q-gram count, $t_{\mathcal{M}}$ [OR-6].

- **New Formal Definitions of Deduplication**

  - The formal definition of the deduplication process is newly introduced (see Definition 36).
  - The definitions of the filtering process (Definition 38) and the matching process are newly introduced and formally expressed using set theory.
  - The concept of a self-similarity join in a similarity space, as a function of filtering and matching, is explicitly detailed in Definition 37 [OR-8].

- **Unsupervised Learnable Q-gram Count Filter**

  - Unsupervised learning of expected parameters for a Q-gram filter, based on the source, to provide faster and more reliable results.

  - Proposed Algorithm 7 of UL-BipartiteJoin for two step probabilistic similarity join, demonstrated in experiments on research data with outperforming state-of-the-art algorithms [OR-8].

- **Record Deduplication using DBSCAN**

  - Proposed Algorithm 14 for detection of duplicated cluster with DBSCAN which is scalable and runs faster due it is log-linear time complexity based on experiments Table 5.10 [OR-8].

Chapter 2

Similarity Space Theory

2.1 State-of-the-Art

Thanks to the desire for a more rigorous mathematical view of the theory being expressed by the author, a commitment to the use of the term "similarity space" is made for the first time, a term that currently lacks a fixed concept, unlike terms such as "metric space" or "partial metrics."

In this chapter, the current state-of-the-art and studies of the similarity space, which only began to develop in the 21st century, are described by the author. The current state-of-the-art is first presented in the section, followed by the undertaking of a new perspective on this theory.

Mathematical space, in a rigorous sense, is a set with some added structure. It's an abstract concept that forms the foundation for various areas in mathematics. Here are some key types of mathematical spaces:

- **Euclidean Space ($\mathbb{R}^n$)** [92]: This is the most familiar type of space, consisting of points that have a defined distance between them. In Euclidean geometry, points are defined in terms of coordinates within n-dimensional space, examples of which include the familiar 2D or 3D spaces.

- **Vector Space** [8], [93], [94]: This is a collection of objects called vectors, which can be added together and multiplied by scalars (numbers). Vector spaces are fundamental in linear algebra and are characterized by properties like vector addition and scalar multiplication.

- **Topological Space** [11], [95]: This is a set of points, along with a set of neighborhoods for each point, satisfying a set of axioms relating points and neighborhoods. Topological spaces are used in topology to study concepts like continuity, compactness, and connectedness without necessarily having a notion of distance.

- **Metric Space** [3], [11], [96],: This is a set where a distance (or metric) is defined between any two points. This concept is used to generalize many of the ideas from Euclidean spaces to more abstract spaces.
- **Hilbert and Banach Spaces** [7], [8], [93]: These are abstract vector spaces with additional structure. Hilbert spaces, used in functional analysis and quantum mechanics, have an inner product that allows lengths and angles to be measured. Banach spaces, also used in functional analysis, are normed vector spaces that are complete with respect to the metric induced by the norm.
- **Manifold** [97]: This is a topological space that locally resembles Euclidean space and is a key concept in geometry and physics, particularly in the theory of relativity.

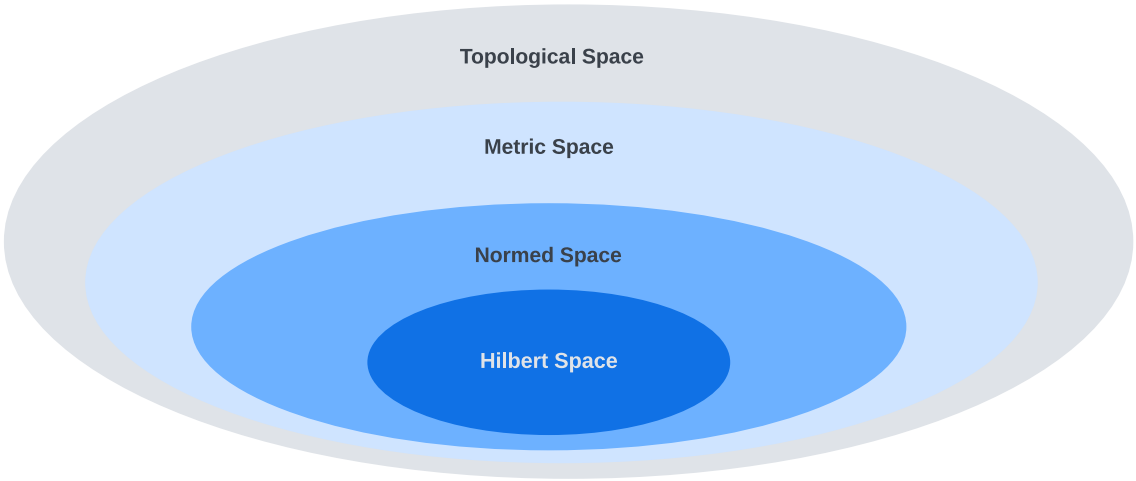

Figure 2.1: Venn Diagram of Mathematical Spaces

Given the variety of mathematical spaces, the author has chosen the one that most closely resembles a similarity space in terms of its axiomatic set, such as metric spaces, partial metric spaces, and the newly conceptualized similarity metric.

2.1.1 Metric Space

The theory of metric space is a well defined mathematical concept. In 1906 Maurice Frechét introduced metric spaces in his work [98] in the context of functional analysis. His main research was to study real-valued functions in metric space in order to generalize the theory of functions of several or even infinitely many variables. The idea was further developed by Felix Hausdorff [99]. Recall the well-known definition.

Definition 1 (Metric Space [7], [11], [96])**.** *Let $\mathcal{X}$ be a non-empty set. Then, a function $d\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a distance metric if for all subsets $x, y, z \in \mathcal{X}$, the following conditions are fulfilled:*

(D1) $d(x, y) = d(y, x)$ (symmetry),

(D2) $d(x, z) \leq d(x, y) + d(y, z)$ (triangle inequality),

(D3) $d(x, y) = 0 \iff x = y$ (identity of indiscernibles).

A metric space is an ordered pair $(\mathcal{X}, d)$.

In addition to these axioms, it is implied that $d(x, y) \geq 0$ (non-negativity) for any $x, y \in \mathcal{X}$. Although this axiom appears redundant, as discussed in [2], [6], its formal derivation can be found in [11].

2.1.2 Partial Metric Space

Partial metric spaces extend the conventional framework of metric spaces by incorporating the notion of self-distance.

Definition 2 (Partial Metric Space [100]–[105])**.** *Let* $\mathcal{X}$ *be a non-empty set. A partial metric or p-metric function* $p\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *is a function such that*

(P1) $p(x, y) = p(y, x)$ (symmetry),

(P2) $p(x, z) + p(y, y) \leq p(x, y) + p(y, z)$ (triangle inequality),

(P3) $p(x, x) = p(x, y) = p(y, y) \iff x = y$ (identity of indiscernibles),

(P4) $p(x, y) \geq 0$ (non-negativity),

(P5) $p(x, y) \geq p(x, x)$ (small self-distance).

Thus, a partial metric space is represented by the ordered pair $(\mathcal{X}, p)$.

This concept was first introduced in foundational works [100], [101] and allows for the possibility of non-zero self-distances, differentiating it from traditional metric spaces. Specifically, a metric space is a special case of a partial metric space where all self-distances are zero, i.e., $p(x, x) = p(y, y) = 0$. Consequently, in such cases, the term $p(y, y)$ in condition (P2) becomes redundant. The introduction of non-zero self-distances was motivated by the need to define a measure of similarity that accommodates distinctions in self-comparison.

2.1.3 Hilbert Space

Definition 3 (Hilbert Space [7], [8], [93])**.** *Let $\mathcal{H}$ be a vector space over the field of real or complex numbers. $\mathcal{H}$ is a Hilbert space if it is equipped with an inner product $\langle \cdot, \cdot \rangle$ and satisfies the following axioms:*

1. ***Vector Space**: $\mathcal{H}$ must satisfy all the properties of a vector space, including vector addition, scalar multiplication, and the existence of a zero vector.*
2. ***Inner Product**: The inner product $\langle x, y \rangle$ for any $x, y \in \mathcal{H}$ must satisfy:*
   - *Conjugate Symmetry: $\langle x, y \rangle = \overline{\langle y, x \rangle}$ (where $\overline{z}$ denotes the complex conjugate of z).*
   - *Linearity in the First Argument: $\langle ax + by, z \rangle = a\langle x, z \rangle + b\langle y, z \rangle$ for all scalars a, b and all $x, y, z \in \mathcal{H}$.*
   - *Positive Definiteness: $\langle x, x \rangle \geq 0$ for all $x \in \mathcal{H}$, and $\langle x, x \rangle = 0$ if and only if $x = 0$.*
3. ***Completeness**: $\mathcal{H}$ must be complete with respect to the norm induced by the inner product. That is, every Cauchy sequence in $\mathcal{H}$ must converge to an element in $\mathcal{H}$. The norm $\|x\|$ is defined as $\sqrt{\langle x, x \rangle}$.*

In a Hilbert space, the concepts of convergence, orthogonality, and projection can be defined analogously to Euclidean spaces, but in an infinite-dimensional setting. Hilbert spaces are essential in various areas of mathematics, including functional analysis and quantum mechanics, where they provide the framework for the study of quantum states and operators.

2.1.4 Similarity Metric

The initial reference to this subject is found in Ma [13]. Chen [21] subsequently presents an axiom system, demonstrating that the sum and product of a similarity result in another similarity. This assertion, which lacks mathematical proof, includes the set intersection, mutual information, and protein sequences analyzed with the Smith-Waterman algorithm [70] as examples of similarity metrics. Their study concentrates on normalizing similarity for sequences, in contrast to the broader approach in this thesis. They establish the triangle inequality for the similarity on set intersections but do not address the duality of construction from a metric space, which this thesis further explores. Moreover, their discussion on the relationship between similarity and distance metrics in Lemma 8 differs in

form and normalization approach from the methods developed in this thesis, particularly in Theorem 26.

Chen [6] and Elzinga [2] identify the general solution for constructing a similarity from a distance as an *unsolved open problem.* This thesis advances this field, especially in Chapter 2.5 and Theorems 12 and 18, proposing a construction based on fundamental set theory principles, which support all theories addressed in this thesis. Within this theory, a general approach for normalizing similarity, termed *Generalized Rozinek Similarity*, is articulated in Theorem 18. This approach, specifically applied to cases of edit similarity and the longest common subsequence (LCS), is further elaborated in Theorem 26, demonstrating the application of the principles of this novel theory.

Unlike in the current state-of-the-art, this thesis, for the first time, shows the context within other mathematical concepts and spaces, embedding of measure space, probability space, and Hilbert space. It introduces novel fixed-point theory in similarity space and a dualistic view in linear regression analysis.

Definition 4 (Similarity Metric [2], [4], [6], [13], [21], [106])**.** *Let $\mathcal{X}$ be a non-empty set. Then, a function $s\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a similarity if for all subsets $x, y, z \in \mathcal{X}$, it satisfies the following conditions:*

(s1) $s(x, y) = s(y, x)$ (symmetry),

(s2) $s(x, x) \geq 0$ (self-similarity non-negativity),

(s3) $s(x, x) \geq s(x, y)$ (bounded by self-similarity),

(s4) $s(x, z) + s(y, y) \geq s(x, y) + s(y, z)$ (triangle inequality),

(s5) $s(x, x) = s(x, y) = s(y, y) \iff x = y$ (identity of indiscernibles).

The condition s1 states that $s(x, y)$ is symmetric. The condition s2 states that for any subset $x \in \mathcal{X}$ the self-similarity is non-negative.

Although it is not mandatory to set this lower bound to zero, it is a common and reasonable choice. Condition s3 says that for any x, the intrinsic similarity is not less than the similarity between x and any y. Condition s5 says that the statements $s(x, x) = s(y, y) = s(x, y)$ and $x = y$ are equivalent.

Condition S4 is equivalent to the triangle inequality under a monotonic convex transformation in metric space. The formation of such a triangle inequality is shown below. However, this thesis presents a significantly broader class of functions, as newly demonstrated in Theorem 10.

Theorem 1 (Triangle Inequality of Similarity Metric [OR-1], [107])**.** *Any decreasing monotonic convex transformation f of the triangular inequality of the metric d forms a triangular inequality of similarity metric s:*

$$d(x,z) \leq d(x,y) + d(y,z) \xrightarrow{f} s(x,z) + s(y,y) \geq s(x,y) + s(y,z), \tag{2.1}$$

Proof. A real-valued function $f(d)$ is said to be convex over the interval $[a,b] \in \mathbb{R}$ if for any $d_1, d_2 \in [a,b]$ and any $\lambda \in [0,1]$, it is held that

$$\lambda f(d_1) + (1-\lambda) f(d_2) \geq f(\lambda d_1 + (1-\lambda) d_2) \tag{2.2}$$

The validity of the triangle inequality $s(x,z)+s(y,y) \geq s(x,y)+s(y,z)$ can be proven from the dual notion of distance $s(x,y) = f(d(x,y))$ by applying $d(x,z) \leq d(x,y)+d(y,z)$ and considering possible cases as follows [107].

Case 1: $d(x,z) \leq d(x,y)$

Thus, it is obtained $f(d(x,z)) \geq f(d(x,y))$. As $0 \leq d(y,z)$, it follows that $f(0) = f(d(y,y)) \geq f(d(y,z))$. Both expressions are summed

$$f(d(y,y)) + f(d(x,z)) \geq f(d(x,y)) + f(d(y,z)). \tag{2.3}$$

So the claim is proven.

Case 2: $d(x,z) \leq d(y,z)$

The reasoning is analogous to the above, just flipping x and z.

Case 3: $d(x,z) > d(x,y) \wedge d(x,z) > d(y,z)$

As a metric is assumed, $d(x,z) \leq d(x,y) + d(y,z)$. Hence

$$1 \leq \frac{d(y,z)}{d(x,z)} + \frac{d(x,y)}{d(x,z)} \implies 0 \leq 1 - \frac{d(y,z)}{d(x,z)} \leq \frac{d(x,y)}{d(x,z)} \leq 1 \tag{2.4}$$

Let pick any $\lambda \in \left[0, \frac{d(x,y)}{d(x,z)}\right]$ such that $1 - \frac{d(y,z)}{d(x,z)} \leq \lambda \leq \frac{d(x,y)}{d(x,z)}$. Obviously $0 \leq \lambda \leq 1$. It is immediately seen that $\lambda d(x,z) \leq d(x,y)$ and $(1-\lambda)d(x,z) \leq d(y,z)$. From the definition of convexity, it is obtained that

$$(1-\lambda) f(0) + \lambda f(d(x,z)) \geq f((1-\lambda)0 + \lambda d(x,z)) \tag{2.5}$$

$$f(\lambda d(x,z)) \geq f(d(x,y)) \tag{2.6}$$

with the last inequality being due to the fact that f is monotonic decreasing. Similarly

$$\lambda f(0) + (1-\lambda) f(d(x,z)) \geq f(\lambda 0 + (1-\lambda) d(x,z)) \tag{2.7}$$

$$f((1-\lambda) d(x,z)) \geq f(d(y,z)) \tag{2.8}$$

By summing all inequalities (2.8) by transitivity $\geq$, it is gotten that

$$f(0) + f(d(x,z)) \geq f(d(x,y)) + f(d(y,z)) \tag{2.9}$$

so obviously the triangle inequality holds here too. □

2.2 Definition of Similarity Space

Especially in the last two decades, the research and development of a formal definition of similarity (or similarity space) has begun. The new applications and the purpose of this thesis call for a general consensus and search for well-defined axiomatic systems and theoretical foundations instead of using a non-intuitive duality with the distance. Based on [2], [6] the author introduces an axiomatic system for a similarity, which is in agreement with the current notions with small change in bounded by self-similarity.

Definition 5 (Similarity Space [2], [4], [13], [OR-1], [OR-3], [OR-7], [OR-8], [108], [OR-9])**.** *Let $\mathcal{X}$ be a nonempty set. A function $s\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ is called a similarity on $\mathcal{X}$ if for any elements $x, y, z \in \mathcal{X}$, the following properties hold:*

(S1) $s(x,y) = s(y,x)$ (symmetry),

(S2) $s(x,z) + s(y,y) \geq s(x,y) + s(y,z)$ (triangle inequality),

(S3) $s(x,x) = s(x,y) = s(y,y)$ *if and only if* $x = y$ (identity of indiscernibles),

(S4) $s(x,y) \geq 0$ (non-negativity),

(S5) $s(x,y) \leq \min\{s(x,x), s(y,y)\}$ (bounded by self-similarity).

A similarity space is an ordered pair $(\mathcal{X}, s)$ such that $\mathcal{X}$ is nonempty set and s is similarity on $\mathcal{X}$.

Compared to the original system, the axiom of bounded self-similarity could be removed, as it can be derived from the remaining axioms; this modification is presented by the author in [OR-1]. However, from a topological standpoint, the uniqueness of limit convergence, which is a common requirement in functional analysis, is not guaranteed by the system.

Theorem 2 (Bounded Self-Similarity)**.** *A similarity satisfies* $s(x, y) \leq \frac{s(x,x)+s(y,y)}{2}$.

Proof. Assuming $z = x$,

$$\begin{aligned} &s(x,x) + s(y,y) \geq s(x,y) + s(y,x) \text{ by triangle inequality} \\ &s(x,x) + s(y,y) \geq 2s(x,y) \text{ by symmetry} \\ &s(x,y) \leq \frac{s(x,x) + s(y,y)}{2} \end{aligned} \tag{2.10}$$

□

By removing another boundary condition of non-negativity, S4, difficulties could be encountered in applying measure theory, which requires non-negativity. On the other hand, such an axiomatic system would be a superset of Hilbert space, making Hilbert space a subspace of the similarity space. The proof is shown in Theorem 14.

A few issues require attention. The term 'similarity metric' is already an established convention. The use of 'metric' should be understood as referring to a monotonously decreasing convex transformation of a partial metric or a distance metric, as will be shown further in the next section. By doing so, misunderstandings can be avoided, and the term will be referred to as 'similarity' only.

Unlike D3, $d(x, x) = 0$, the similarity has an upper bound defined by in $\min\{s(x, x) + s(y, y)\}$ and allows $s(x, x) \neq s(y, y)$. At first sight, this may seem counter-intuitive: x is more (or less) similar to itself than y. In spatial considerations of dissimilarity and distance, this does not arise since $d(x, x) = 0$ for all objects. Similarity depends on the set of common features, and the result is the possibility of non-identical self-similarities. If common features are interpreted as 'description lengths' or 'complexities', unequal self-similarities become quite natural, and if x has more features than y, then $s(x, x) > s(y, y)$ is observed [2]. For instance, having a German word x = 'Einkommensteuererklärung' (income tax return) and y = 'Steuer' (tax), then $s(x, x) \geq s(y, y)$ when counting common characters or Q-grams.

The author suggests in addition having non-negativity in S4 because the similarity doesn't have a direction—in contrast to a vector, it is a scalar value, and so it doesn't make sense to assign to it a negative sign, similar to non-negativity in a metric space. The same principle should be valid for a similarity as a requirement for "symmetric measurement". The distance between objects is observed to remain the same when measured from another direction. The second reason follows from measure theory, where there is a non-negativity condition $\mu(x) \geq 0$ for a measure μ on the set $\mathcal{X}$ [109], [110].

Theorem 3 (Linear Transformation)**.** *Every positive linear transformation* $T_L\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ *of a similarity is a similarity:*

$$s_L(x,y) = T_L(s(x,y)) = \alpha s(x,y) + \beta \tag{2.11}$$

where $\alpha, \beta \in \mathbb{R}$ *and* $\alpha > 0, \beta \geq 0$*.*

Proof. Let $\overline{s}(x,y)$ be a positive linear transformation of $s(x,y)$ such that $\overline{s}(x,y) = \alpha s(x,y) + \beta$ for $\alpha > 0$ and $\beta \geq 0$.

S1. By symmetry by multiplication α and adding β

$s(x,y) = s(y,x) \implies \alpha s(x,y) + \beta = \alpha s(y,x) + \beta \implies \overline{s}(x,y) = \overline{s}(y,x)$

S2. By the triangle inequality, it is obtained

$$\begin{aligned} &s(x,z) + s(y,y) \geq s(x,y) + s(y,z) \\ &\alpha(s(x,z) + s(y,y)) + 2\beta \geq \alpha(s(x,y) + s(y,z)) + 2\beta \\ &\overline{s}(x,z) + \overline{s}(y,y) \geq \overline{s}(x,y) + \overline{s}(y,z) \end{aligned} \tag{2.12}$$

By multiplication by α and adding 2β, the proof is complete. Similarly, in cases in cases S3, S4 and S5 the procedure is followed. □

This theorem allows me to apply any linear standardization or re-scaling without any violations of the axioms. In statistics, there is very often used a standard score

$$X' = \frac{X - \mu_s}{\sigma_s}, \tag{2.13}$$

where μ_s is the mean and σ_s is the standard deviation. Another example could be taken from min-max feature scaling

$$X' = a + \frac{(X - X_{min})(b-a)}{X_{max} - X_{min}}, \tag{2.14}$$

where X_{min} denotes the minimum value, and X_{max} the maximum value. All values are re-scaled (normalization) to lie within the range $[a,b]$. When the parameters $a = 0$, $b = 1$ are chosen, then this is a unity-based normalization.

For instance, it should be clear that two errors in a comparison of short strings are more critical than in a comparison of long strings. Therefore, it is necessary in some circumstances to normalize the similarity. Until the beginning of this century, no such normalization preserving the metric axioms was known for the edit distance metric. Initially, [111] developed a normalized edit distance metric, with the range $[0,1]$. It is obvious that for any normalized distance metric $d_n(x,y)$, there is also a normalized similarity $s_n(x,y) = 1 - d_n(x,y)$ satisfying Definition 5.

Because this axiomatic system is considered too general and valid for any unnormalized similarity functions, a new specific axiomatic system for a normalized similarity in the range $[0, 1]$ is introduced for this case.

Definition 6 (Normalized similarity)**.** *A function* $s_n(x, y) \colon \mathcal{X} \times \mathcal{X} \to [0, 1] \subset \mathbb{R}$ *is a normalized similarity if, such that for all subsets* $x, y, z \in \mathcal{X}$*, it satisfies the following conditions:*

(N1) $s_n(x, y) = s_n(y, x)$ (symmetry),

(N2) $s_n(x, z) + 1 \geq s_n(x, y) + s_n(y, z)$ (triangle inequality),

(N3) $s_n(x, y) = 1 \iff x = y$ (identity of indiscernibles),

(N4) $s_n(x, y) \geq 0$ (non-negativity).

A normalized similarity space is an ordered pair $(\mathcal{X}, s_n)$*.*

The axiom related to bounded by self-similarity (S5) is relaxed compared to Definition 5; however, a stricter, meaningful special case of that definition is created by enforcing $s_n(x, x) = 1$, which also represents the least upper bound of $s_n(x, y)$. Due to this normalization, self-similarity is always bounded by the same number $s_n(x, x) = s_n(y, y) = 1$. The total dissimilarity defines the greatest lower bound $s_n(x, y) = 0$. Thus, the requirements for both limit conditions N3 and N4 stretch the similarity to its boundaries.

The boundedness by self-similarity could be derived from the rest of the axioms.

Theorem 4 (Bounded by Self-similarity)**.** *A normalized similarity satisfies* $s_n(x, y) \leq 1$*.*

Proof. By identity of indiscernibles (N3) is a least upper bound given

$$s_n(x, y) = s_n(y, y) = s_n(x, x) = 1 \tag{2.15}$$

and Theorem 2 implies

$$s_n(x, y) \leq \frac{s_n(x, x) + s_n(y, y)}{2} = 1 \tag{2.16}$$

□

With these properties, a connection is also made with probability theory, where it is ensured that the probability of similarity is bounded by $0 \leq P(x, y) \leq 1$ and similarly, $0 \leq s_n(x, y) \leq 1$.

Theorem 5 (Convex Combinations)**.** *A convex combination* $T_C \colon \mathbb{R} \to \mathbb{R}$ *of normalized similarities is again a normalized similarity:*

$$s_{n_C}(x, y) = T_C(s_n(x, y)) = \sum_{i=1}^{m} \alpha_i s_i = \alpha_1 s_1 + \alpha_2 s_2 + \ldots + \alpha_m s_m \tag{2.17}$$

where $\sum_{i=1}^{m} \alpha_i = 1$ *and* $0 \leq \alpha_i \leq 1$.

Proof. The proof is continued for each axiom, assuming $\overline{s}(x,y) = \sum_{i=1}^{m} \alpha_i s_i(x_i, y_i)$,

N1. It is obvious that

$$\begin{aligned} \sum_{i=1}^{m} \alpha_i s_i(x_i, y_i) &= \sum_{i=1}^{m} \alpha_i s_i(y_i, x_i) \\ \overline{s}(x,y) &= \overline{s}(y,x) \end{aligned} \tag{2.18}$$

Similarly, N2, N3 and N4 are trivial. □

This property of convex combinations allows to assemble different normalized similarities together and obtain again a normalized similarity.

2.3 Topology

Topology explores the properties of space that are preserved under continuous transformations. Further study will reveal the importance of space separation and limit uniqueness, key concepts in the field of functional analysis.

Theorem 6 (Induced Elementary Metric)**.** *If* $s(x,y)$ *is a similarity on* $\mathcal{X}$*, then the function* $d^s \colon X \times X \to \mathbb{R}_{\geq 0}$ *given by*

$$d^s(x,y) = s(x,x) + s(y,y) - 2s(x,y) \tag{2.19}$$

is induced elementary metric on $\mathcal{X}$*.*

Proof. Consider $x, y \in \mathcal{X}$. Then $d^s(x,y) = s(x,x) + s(y,y) - 2s(x,y)$ is always non-negative by the bounded self-similarity (S5) because $s(x,y) \leq \min\{s(x,x), s(y,y)\}$ holds. Moreover, if $d^s(x,y) = d^s(y,x) = 0$ it follows that $x = y$ because $s(x,x) = s(x,y) = s(y,y)$. Furthermore, the triangular inequality holds

$$\begin{aligned} d^s(x,y) &= s(x,x) + s(y,y) - 2s(x,y) \\ &\leq s(x,x) + s(y,y) - 2[s(x,z) + s(y,z) - s(z,z)] \\ &= [s(x,x) + s(z,z) - 2s(x,z)] + [s(y,y) + s(z,z) - 2s(y,z))] \\ &= d^s(x,z) + d^s(y,z). \end{aligned}$$

□

Lemma 1 (Induced Quasi-Metric)**.** *If $(\mathcal{X}, s)$ is a similarity space, then the function $d_q^s\colon X \times X \to \mathbb{R}_{\geq 0}$ defined by*

$$d_q^s(x, y) = s(x, x) - s(x, y), \tag{2.20}$$

is a quasi-metric on $\mathcal{X}$.

Proof. Consider $x, y, z \in \mathcal{X}$, then $d_q^s(x, y) = s(x, x) - s(x, y)$ is always non-negative because $s(x, y) \leq s(x, x)$ is given by (S5). It is obvious that $d_q^s(x, y) = d_s(y, x) = 0 \iff x = y$. Finally

$$\begin{aligned} d_q^s(x, y) &= s(x, x) - s(x, y) \\ &\leq s(x, x) - s(x, z) + s(z, z) - s(z, y) \\ &= d_q^s(x, z) + d_q^s(z, y). \end{aligned} \tag{2.21}$$

□

If d_q^s is a quasi-metric on $\mathcal{X}$, then the function d^s defined on $\mathcal{X} \times \mathcal{X}$ by $d^s(x, y) = \max\{d_q^s(x, y), d_q^s(y, x)\}$, is a metric on $\mathcal{X}$ [103].

Lemma 2 (Induced Elementary Similarity)**.** *Given a similarity s and a metric d on $\mathcal{X}$, then for every $x, y \in \mathcal{X}$ there exists an induced similarity*

$$s(x, y) = \frac{s(x, x) + s(y, y) - d(x, y)}{2}. \tag{2.22}$$

Proof. By expressing $s(x, y)$ from Theorem 6 and substitution $d^s(x, y) = d(x, y)$. □

Remark 1. *Let be a measure space as a triple $(\mathcal{X}, \mathcal{F}, \mu)$, where $\mathcal{X}$ is non-empty set, $\mathcal{F}$ is a σ-algebra on the set $\mathcal{X}$ and μ is a measure on $(\mathcal{X}, \mathcal{F})$. Consider subsets $x, y \in \mathcal{F}$. Without loss of generality, let denote $s(x, x) = \mu(x)$, $s(y, y) = \mu(y)$ and $d(x, y) = \mu(x \triangle y)$. Then the induced elementary similarity in the measure space of the form is obtained*

$$\begin{aligned} s(x, y) &= \frac{s(x, x) + s(y, y) - d(x, y)}{2} = \frac{\mu(x) + \mu(y) - \mu(x \triangle y)}{2} \\ &= \frac{\mu(x) + \mu(y) + \mu(x \cap y) - \mu(x \cup y)}{2} \\ &= \frac{2(\mu(x) + \mu(y) - \mu(x \cup y))}{2} \\ &= \mu(x \cap y). \end{aligned} \tag{2.23}$$

As a result, a measure on the intersection of subsets $\mu(x \cap y)$ is obtained, which is also consistent with the intuitive notion that similarity depends on measuring common elements of sets [OR-1].

Definition 7 (Open s-Ball and Closed s-Ball)**.** *Let $(\mathcal{X}, s)$ be similarity space, and let $x \in \mathcal{X}$ and $\epsilon \geq 0$. The open s-ball of radius ϵ with centre $\mathcal{X}$ is the set*

$$\mathcal{B}_s(x, \epsilon) = \{y \in \mathcal{X} \colon s(x, y) > \max\{s(x, x), s(y, y)\} - \epsilon\}. \tag{2.24}$$

The closed s-ball of radius ϵ with centre $\mathcal{X}$ is the set

$$\overline{\mathcal{B}}_s(x, \epsilon) = \{y \in \mathcal{X} \colon s(x, y) \geq \max\{s(x, x), s(y, y)\} - \epsilon\}. \tag{2.25}$$

Theorem 7. *The collection $\{\mathcal{B}_s(x, \epsilon) \mid x \in \mathcal{X}, \epsilon > 0\}$ of all open s-balls in a similarity space $(\mathcal{X}, s)$ form a basis for topology τ.*

Proof. Since $x \in \mathcal{B}_s(x, \epsilon)$ for all $x \in \mathcal{X}$ and $\epsilon > 0$, it follows at once that the open s-balls cover all of $\mathcal{X}$.

Next, it is supposed that $\epsilon_1, \epsilon_2 > 0, y, z \in X, z \in \mathcal{B}_s(z, \epsilon) \subseteq \mathcal{B}_s(x, \epsilon_1) \cap \mathcal{B}_s(y, \epsilon_2)$ for $\epsilon = \min\{s(x, z) + \epsilon_1 - s(z, z), s(y, z) + \epsilon_2 - s(z, z)\} = \min\{\epsilon_1 + s(x, z), \epsilon_2 + s(y, z)\} - s(z, z)$. For any $w \in \mathcal{X}$, if $s(w, z) > s(w, w) - \epsilon$, then by the triangle inequality (S2), it is obtained that

$$\begin{aligned} s(w, x) &\geq s(w, z) + s(z, x) - s(z, z) \\ &> (s(w, w) - \epsilon) + s(z, x) - s(z, z) \\ &= s(w, w) - (\epsilon_1 + s(x, z) - s(z, z)) + s(z, x) - s(z, z) \\ &= s(w, w) - \epsilon_1, \end{aligned} \tag{2.26}$$

which implies $w \in \mathcal{B}_s(x, \epsilon_1)$. Therefore, $\mathcal{B}_s(z, \epsilon) \subseteq \mathcal{B}_s(x, \epsilon_1)$. Similarly, it can be shown that $\mathcal{B}_s(z, \epsilon) \subseteq \mathcal{B}_s(y, \epsilon_2)$, completing the proof. □

The following definition is now justified.

Definition 8. *Given a similarity space $(\mathcal{X}, s)$, the* induced topology *on the non-empty set $\mathcal{X}$ is the topology τ_s generated by the open s-balls in $\mathcal{X}$.*

Definition 9. *Let $(\mathcal{X}, s)$ be similarity space. All open s-balls in $\mathcal{X}$ are open sets, and a subset $A \subseteq X$ is open in the induced topology when for all $x \in A$ there exists an $\epsilon > 0$ such that $\mathcal{B}_s(x, \epsilon) \subseteq A$.*

Definition 10. *Any topological space (X, τ_s) for which there exists a similarity s on $\mathcal{X}$ inducing the given topology τ_s is called a* similarizable space.

Theorem 8. *Every similarizable space is Hausdorff space $\mathcal{T}_2$.*

Proof. Consider a topological space $\mathcal{X}$ with a topology induced by a similarity function s. For any two distinct elements $x, y \in \mathcal{X}$, without loss of generality, assume that $s(x,x) \leq s(y,y)$. Should this not be the case, x and y could be interchanged.

If $s(x,x) = s(x,y)$ is given, then by the identity of indiscernibles property (S3), it would imply $x = y$, which contradicts the assumption that $\mathcal{X}$ and y are distinct. Hence, it is concluded that $s(x,y) < s(x,x) \leq s(y,y)$.

Now, define $\epsilon = \frac{1}{2}(s(x,x) - s(x,y)) > 0$. It is claimed that the open s-balls $\mathcal{B}_s(x,\epsilon)$ and $\mathcal{B}_s(y,\epsilon)$ separate $\mathcal{X}$ and y. To see this, suppose $z \in \mathcal{B}_s(x,\epsilon) \cap \mathcal{B}_s(y,\epsilon)$ is given. Then, it would be found that

$$\begin{aligned} s(x,y) + s(z,z) &\geq s(x,z) + s(z,y) \\ &> (s(x,x) - \epsilon) + (s(z,z) - \epsilon) \\ &= s(x,x) + s(z,z) - \frac{2(s(x,x) - s(x,y))}{2} \\ &= s(x,y) + s(z,z), \end{aligned} \tag{2.27}$$

which is a contradiction. Therefore, such an element z cannot exist, and the open s-balls are indeed disjoint, i.e., $\mathcal{B}_s(x,\epsilon) \cap \mathcal{B}_s(y,\epsilon) = \emptyset$. This shows that every similarizable space is a Hausdorff space $\mathcal{T}_2$. □

The uniqueness of limit points for convergent sequences in a similarity space can be established from the above Theorem 8. This is because every similarity space is a Hausdorff space, and in every Hausdorff space, a convergent sequence has at most one limit in $\mathcal{X}$.

Lemma 3. *If $(\mathcal{X}, s)$ is a similarity space, then the function $d^s \colon X \times X \to \mathbb{R}_{\geq 0}$ defined by $d^s(x,y) = s(x,x) + s(y,y) - 2s(x,y)$ is a metric on $\mathcal{X}$ such that $\mathcal{T}_2(s) = \mathcal{T}_2(d^s)$.*

Proof. This is obvious from Lemma 2 and Theorem 8. □

2.4 Convergence and Continuity

Exploring converging sequences, Cauchy sequences, and the principles of continuity and convergence is crucial for grasping the completeness of spaces, a key factor for ensuring stability and predictability of limits in functional analysis.

Definition 11. *Let $(\mathcal{X}, s)$ be a similarity space. Then*

(i) A sequence $\{x_n\}_{n=1}^{\infty}$ in a similarity space $(\mathcal{X}, s)$ converges to an element $x \in \mathcal{X}$ if and only if

$$\lim_{n\to\infty} s(x_n, x_n) = \lim_{n\to\infty} s(x_n, x) = s(x,x). \tag{2.28}$$

(ii) A sequence $\{x_n\}_{n=1}^{\infty}$ *in a similarity space* $(\mathcal{X}, s)$ *is called a Cauchy sequence if there exist*

$$\lim_{n,m\to\infty} s(x_n, x_m) = s(x, x). \tag{2.29}$$

(iii) A similarity space $(\mathcal{X}, s)$ *is said to be complete if every Cauchy sequence* $\{x_n\}_{n=1}^{\infty}$ *with respect to* τ_s*, to an element* $x \in \mathcal{X}$ *has a limit* $\lim_{n,m\to\infty} s(x_n, x_m) = s(x, x)$ *that is also in* $\mathcal{X}$*.*

Lemma 4. *Let* $(\mathcal{X}, s)$ *be a similarity space. Then, the following claims can be made:*

(i) A sequence $\{x_n\}_{n=1}^{\infty}$ *is a Cauchy sequence in the similarity space* $(\mathcal{X}, s)$ *if and only if it is a Cauchy sequence in the metric space* $(\mathcal{X}, d^s)$*.*

(ii) A similarity space $(\mathcal{X}, s)$ *is complete if and only if the metric space* $(\mathcal{X}, d^s)$ *is complete. Furthermore,*

$$\lim_{n\to\infty} d^s(x_n, x) = 0 \iff s(x, x) = \lim_{n\to\infty} s(x_n, x) = \lim_{n,m\to\infty} s(x_n, x_m). \tag{2.30}$$

Proof. Claim (i)

It is first demonstrated that every Cauchy sequence in $(\mathcal{X}, s)$ is a Cauchy sequence in $(\mathcal{X}, d^s)$. To this end let $\{x_n\}_{n=1}^{\infty}$ be a Cauchy sequence in $(\mathcal{X}, s)$. Then there exists $a \in \mathbb{R}_{\geq 0}$ such that, given $\epsilon > 0$, there is $N \in \mathbb{N}$ with $|s(x_n, x_m) - a| < \frac{\epsilon}{2}$ for all $n, m \geq N$. By applying the triangle inequality for the quasi-metric d_q^s and from the definition of the quasi-metric, it is obtained that

$$\begin{aligned} d_q^s(x_n, x_m) &= s(x_n, x_n) - s(x_n, x_m) \\ &= |s(x_n, x_n) - a + a - s(x_n, x_m)| \\ &\leq |s(x_n, x_n) - a| + |a - s(x_n, x_m)| \\ &< \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon \end{aligned} \tag{2.31}$$

for all $n, m \geq N$. The same argument applies to $d_q^s(x_m, x_n)$ for all $n, m \geq N$. It is observed that $\{x_n\}_{n=1}^{\infty}$ is indeed a Cauchy sequence in the metric space $(\mathcal{X}, d^s)$, since $d^s(x_n, x_m) = d^s(x_m, x_n) = \max\{d_q^s(x_n, x_m), d_q^s(x_m, x_n)\}$.

Claim (ii)

($\Rightarrow$) Completeness of $(\mathcal{X}, d^s)$ implies completeness of $(\mathcal{X}, s)$:
Suppose that $(\mathcal{X}, d^s)$ is complete. Then, it is necessary to demonstrate that $(\mathcal{X}, s)$ is also complete. Given a Cauchy sequence $\{x_n\}_{n=1}^{\infty}$ in $(\mathcal{X}, s)$, it is also a Cauchy sequence in $(\mathcal{X}, d^s)$ due to the prior claim. Since the similarity space $(\mathcal{X}, d^s)$ is complete, it is deduced

that there exists $y \in \mathcal{X}$ such that $\lim_{n\to\infty} s(x_n, x_n) = s(y, y)$. Let $\epsilon > 0$ then there exists $n \in \mathbb{N}$ such that $d^s(y, x_n) < \frac{\epsilon}{2}$ whenever $n \geq N$. Thus

$$\begin{aligned} |s(y,y) - s(x_n,x_n)| &= |s(y,y) - s(y,x_n) + s(y,x_n) - s(x_n,x_n)| \\ &\leq |s(y,y) - s(y,x_n)| + |s(y,x_n) - s(x_n,x_n)| \\ &= d_q^s(y,x_n) + d_q^s(x_n,y) \\ &< \max\{d_q^s(y,x_n), d_q^s(x_n,y)\} + \max\{d_q^s(x_n,y), d_q^s(y,x_n)\} \\ &= 2d^s(y,x_n) < \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon \end{aligned} \tag{2.32}$$

whenever $n \geq N$. This shows that $(\mathcal{X}, s)$ is complete.

($\Leftarrow$) Completeness of $(\mathcal{X}, s)$ implies completeness of $(\mathcal{X}, d^s)$:
Now, the aim is to show the converse, that every Cauchy sequence $\{x_n\}_{n=1}^{\infty}$ in $(\mathcal{X}, d^s)$ is a Cauchy sequence in $(\mathcal{X}, s)$. Given $\epsilon = \frac{1}{2}$, then there exists $n \in \mathbb{N}$ such that $d^s(x_n, x_m) < \frac{1}{2}$ for all $n, m \geq N$. Since

$$d_q^s(x_n, x_N) - s(x_n, x_n) = d_q^s(x_N, x_n) - s(x_N, x_N) \tag{2.33}$$

then

$$\begin{aligned} |s(x_n,x_n)| &= |-d_q^s(x_N,x_n) + s(x_N,x_N) + d_q^s(x_n,x_N)| \\ &\leq d_q^s(x_N,x_n) + s(x_N,x_N) + d_q^s(x_n,x_N) \\ &\leq 2\max\{d_q^s(y,x_n), d_q^s(x_n,y)\} + s(x_N,x_N) \\ &= 2d^s(x_N,x_n) + s(x_N,x_N) \\ &\leq 1 + s(x_N,x_N) \end{aligned} \tag{2.34}$$

So, the sequence $\{s(x_n, x_n)\}_{i=n}^{\infty}$ is bounded in $\mathbb{R}_{\geq 0}$, and there exists a subsequence $\{s(x_{n_k}, x_{n_k})\}_{k=1}^{\infty}$ that converges to some real number $a \in \mathbb{R}_{\geq 0}$, formally $\lim_{k\to\infty} s(x_{n_k}, x_{n_k}) = a$.

Next, it is shown that $\{s(x_n, x_n)\}_{n=1}^{\infty}$ is a Cauchy sequence in $\mathbb{R}_{\geq 0}$. Since $\{s(x_n, x_n)\}_{i=n}^{\infty}$ is a Cauchy sequence in $(\mathcal{X}, d^s)$, given $\epsilon > 0$, there exists $n \in \mathbb{N}$ such that $d^s(x_n, x_m) < \frac{\epsilon}{2}$ for all $n, m \geq N$. Thus, for all $n, m \geq N$,

$$\begin{aligned} |s(x_n,x_n) - s(x_m,x_m)| &= |d_q^s(x_n,x_m) - d_q^s(x_m,x_n)| \\ &\leq 2d^s(x_m,x_n) < \epsilon \end{aligned} \tag{2.35}$$

because of

$$s(x_n,x_n) = -d_q^s(x_m,x_n) + s(x_m,x_m) + d_q^s(x_n,x_m). \tag{2.36}$$

Therefore $\lim_{n\to\infty} s(x_n, x_n) = a$. On the other hand,

$$\begin{aligned} |a - s(x_n, x_m)| &= |a - s(x_n, x_n) + s(x_n, x_n) - s(x_n, x_m)| \\ &\leq |a - s(x_n, x_n)| + d_q^s(x_n, x_m) < \epsilon \end{aligned} \tag{2.37}$$

for all $n, m \geq N$. Hence $\lim_{n,m\to\infty} s(x_n, x_m) = a$ and $\{x_n\}_{i=n}^{\infty}$ is a Cauchy sequence in $(\mathcal{X}, s)$.

It follows that the sequence $\{x_n\}_{n=1}^{\infty}$ in $(\mathcal{X}, d^s)$ is a Cauchy sequence in $(\mathcal{X}, s)$ and it converges to an element $y \in \mathcal{X}$ with

$$\lim_{n,m\to\infty} s(x_n, x_m) = \lim_{n\to\infty} s(y, x_n) = s(y, y). \tag{2.38}$$

Then, given ϵ, there exists $n \in \mathbb{N}$ such that

$$s(y, y) - s(y, x_n) < \epsilon \wedge s(y, y) - s(x_n, x_n) < \epsilon \tag{2.39}$$

whenever $n \geq N$. As a consequence, it is concluded that

$$d_q^s(y, x_n) = s(y, y) - s(y, x_n) < \epsilon, \tag{2.40}$$

and

$$\begin{aligned} d_q^s(x_n, y) &= s(x_n, x_n) - s(y, x_n) \\ &\leq |s(y, y) - s(y, x_n)| + |s(y, y) - s(x_n, x_n)| < 2\epsilon \end{aligned} \tag{2.41}$$

whenever $n \geq N$. Therefore, the derived metric space $(\mathcal{X}, d^s)$ is complete. And it can easily be verified that $\lim_{n\to\infty} d^s(x, x_n) = 0$ if and only if $s(x, x) = \lim_{n\to\infty} s(x, x_n) = \lim_{n,m\to\infty} s(x_n, x_m)$ [100], [103]. □

Definition 12 (s-Continuity)**.** *Let $(\mathcal{X}, s_{\mathcal{X}})$ and $(\mathcal{Y}, s_{\mathcal{Y}})$ be similarity spaces. A function $f\colon \mathcal{X} \to \mathcal{Y}$ is called* s-continuous *at a point $a \in \mathcal{X}$ if for every $\epsilon > 0$ there exists $\delta > 0$ such that for all $x \in \mathcal{X}$, if*

$$\begin{aligned} &s_{\mathcal{X}}(x, a) > \max\{s_X(x, x), s_X(a, a)\} - \delta \\ &\implies s_{\mathcal{Y}}(f(x), f(a)) > \max\{s_{\mathcal{Y}}(f(x), f(x)), s_{\mathcal{Y}}(f(a), f(a))\} - \epsilon. \end{aligned} \tag{2.42}$$

Furthermore, a function $f\colon \mathcal{X} \to \mathcal{Y}$ is said to be s-continuous if it is continuous at every point $a \in \mathcal{X}$. In terms of open s-balls, the definition says that $f(\mathcal{B}_s(a, \delta)) \subseteq \mathcal{B}_s(f(a), \epsilon)$.

Definition 13 (Uniform s-Continuity)**.** *Let $(\mathcal{X}, s_{\mathcal{X}})$ and $(\mathcal{Y}, s_{\mathcal{Y}})$ be similarity spaces. A function $f\colon \mathcal{X} \to \mathcal{Y}$ is said to be* uniformly s-continuous *if for every $\epsilon > 0$ there exists*

$\delta > 0$ such that for all $x, y \in \mathcal{X}$, if

$$\begin{aligned} & s_X(x,y) > \max\{s_X(x,x), s_X(y,y)\} - \delta \\ & \implies s_Y(f(x),f(y)) > \max\{s_Y(f(x),f(x)), s_Y(f(y),f(y))\} - \epsilon. \end{aligned} \tag{2.43}$$

Let (X, s) be a similarity space, where $\mathcal{X}$ is a set and $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a similarity function. The aim is to introduce a vector space structure on $\mathcal{X}$ itself. For this purpose, it is assumed that $\mathcal{X}$ can be endowed with such a structure.

Let $\mathcal{X}$ be a vector space over field $\mathbb{F}$ and consider a function $f : \mathcal{X} \to \mathbb{F}$. Assume that $\mathcal{X}$ is equipped with a topology that allows for the concept of limits.

The derivative of f at a point $x \in \mathcal{X}$ is defined in the following manner.

Definition 14 (s-Derivative)**.** *Let $(\mathcal{X}, s_\mathcal{X})$ and $(\mathcal{Y}, s_\mathcal{Y})$ be similarity spaces, and let $f : \mathcal{X} \to \mathcal{Y}$ be a function between these spaces. The s-derivative of f at a point $a \in \mathcal{X}$, denoted by $D_s f(a)$, is defined as*

$$D_s f(a) = \lim_{x \to a} \frac{\max\{s_Y(f(x),f(x)), s_Y(f(a),f(a))\} - s_Y(f(x),f(a))}{\max\{s_X(x,x), s_X(a,a)\} - s_X(x,a)}, \tag{2.44}$$

where $s_\mathcal{X} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ and $s_\mathcal{Y} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ are the similarities in $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Theorem 9 (Sequential Criterion for s-Continuity)**.** *Let $f : \mathcal{X} \to \mathcal{Y}$ be a function between similarity spaces $(\mathcal{X}, s_\mathcal{X})$ and $(\mathcal{Y}, s_\mathcal{Y})$. The function f is s-continuous at a point $x \in \mathcal{X}$ if and only if for every sequence of elements $\{x_n\}_{n=1}^\infty$ in $\mathcal{X}$ converging to $\mathcal{X}$, the sequence $\{f(x_n)\}_{n=1}^\infty$ in $\mathcal{Y}$ converges to $f(x)$.*

Proof. ($\Rightarrow$) Assume that f is continuous at $x \in \mathcal{X}$ and let $\{x_n\}_{n=1}^\infty$ be a sequence of elements in $\mathcal{X}$ converging to $\mathcal{X}$. Let $\epsilon > 0$ be arbitrary. Expressed in s-open balls, there exists $\delta > 0$ such that $f(y) \in B_{s_Y}(f(x), \epsilon)$ for all $y \in B_{s_X}(x, \delta)$. Since $\{x_n\}_{n=1}^\infty$ converges to $\mathcal{X}$, there exists $N \in \mathbb{N}$ such that $x_n \in B_{s_X}(x, \delta)$ for all $n \geq N$. Therefore, for $n \geq N$, it is observed that $f(x_n) \in B_{s_Y}(f(x), \epsilon)$. Given that ϵ was arbitrary, this observation implies that $\{f(x_n)\}_{n=1}^\infty$ converges to $f(x)$.

($\Leftarrow$) Conversely, suppose that f is not continuous at $\mathcal{X}$. Then there exists ϵ^* such that for every $\delta > 0$ there exists $y \in B_{s_X}(x, \delta)$ with $f(y) \notin B_{s_Y}(f(x), \epsilon^*)$. Let $\delta_n = \frac{1}{n}$. For each n, pick $x_n \in B_{s_X}(x, \delta_n)$ such that $f(x_n) \notin B_{s_Y}(f(x), \epsilon^*)$. Then, $\{x_n\}_{n=1}^\infty$ converges to $\mathcal{X}$. However, the sequence $\{f(x_n)\}_{n=1}^\infty$ does not converge to $f(x)$, contradicting the assumption. Therefore, f must be s-continuous at $\mathcal{X}$. □

2.5 Duality of Similarity and Metric Space

The relationship between distance and similarity is not obvious, as distance derives from spatial considerations and similarity relations derive from considering common and non-common features [2], [OR-1]. In many cases, distance is used to measure similarity, although this is far from intuitive and it is often a non-trivial task to find such a dual notion.

In the construction of a similarity space, monotonically decreasing mappings of metric spaces are first examined. These mappings are employed to represent the similarity space, though it is essential to note that the metric is not preserved in this process. Instead, an alternative axiomatic system shapes the resulting similarity space, allowing for the definition and analysis of similarity functions within this new mathematical structure.

Conversely, the metric space can be constructed using the inverse mapping to the similarity space. Above all, it has been shown that certain coefficients and indices (such as the Jaccard index, Tanimoto coefficient, etc.) lie in the similarity space, for which no mathematical space had been assigned before [OR-1].

A more general class of functions that transform a metric space into a dual similarity space is presented, as previously mentioned in [OR-1], [107]. Instead of requiring a decreasing monotonic convex transformation f, as demonstrated in Theorem 1, it is proved that a continuous, strictly decreasing function f is sufficient.

Theorem 10 (Duality of Metric and Similarity Space)**.** *Let $(\mathcal{X}, d)$ be a metric space where $d\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ is a metric, and let $f\colon \mathbb{R}_{\geq 0} \to [a, b]$ be a continuous, strictly decreasing function such that $b = f(0) > 0$ and $\lim_{n\to\infty} f(n) = a \geq 0$. Then, the function $s\colon \mathcal{X} \times \mathcal{X} \to [a, b]$, defined by*

$$s(x, y) = f(d(x, y)), \tag{2.45}$$

for all $x, y \in \mathcal{X}$, forms a similarity space $(\mathcal{X}, s)$.

Proof. Consider arbitrary elements x, y, z from the set $\mathcal{X}$. The distances $d(x, y)$, $d(y, z)$, and $d(x, z)$ are formed.

Define the partially ordered set (poset) $(\mathcal{P}, \leq) = \{d(x, y), d(y, z), d(x, z)\}$. In this poset, the elements are the distances, and the order relation is the usual order of real numbers. The set $\mathcal{P}$ is considered as an antichain, where no order is imposed on the elements relative to each other.

Consider all linear order extensions $E(\mathcal{P})$ of the poset $\mathcal{P}$. A linear order extension of a poset is a linear (total) order that extends the partial order. Given $\mathcal{P}$ is an antichain, there are $|E(\mathcal{P})| = 3! = 6$ possible linear order extensions, corresponding to all permutations of $d(x, y)$, $d(y, z)$, and $d(x, z)$.

Introduce the function $f\colon \mathbb{R}_{\geq 0} \to [a, b]$ that is strictly monotonously decreasing. For two real numbers a and b, where $a < b$, it is observed that $f(a) > f(b)$. This function reverses the order of its inputs.

Apply f to each element in each linear order extension in $E(\mathcal{P})$. This application reverses the order of distances in each linear ordering. For instance, if a linear order extension of $\mathcal{P}$ is $d(x, z) \leq d(x, y) \leq d(y, z)$, applying f yields $f(d(x, z)) \geq f(d(x, y)) \geq f(d(y, z))$. As f is continuous and monotonously decreasing, and $d(y, y) = 0$, it follows that $f(d(y, y)) = b \geq f(d(y, z))$. Summing these inequalities, the following is obtained:

$$f(d(x, z)) + f(d(y, y)) \geq f(d(x, y)) + f(d(y, z)), \tag{2.46}$$

being the triangle inequality for the similarity space for this specific order.

This reasoning is systematically applied to each of the six permutations in $E(\mathcal{P})$, ensuring the triangle inequality for similarity is satisfied for all possible orderings of distances.

The properties of symmetry (S1), identity of indiscernibles (S3), non-negativity (S4), and bounded self-similarity (S5) are follows directly from the properties of the metric d and the definition of the function f. Therefore, the pair $(\mathcal{X}, s)$ indeed forms a similarity space in the sense of Definition 5. □

The range $[a, b]$ is not required to be a closed set. It might be denoted that $a = \inf s(x, y)$ and $b = \sup s(x, y)$ are determined. The introduction of the condition $a \geq 0$ is made to preserve symmetry with the non-negativity of the values of the distance metric $d(x, y)$. When a distortion of a metric space into a similarity space is allowed, it is not necessarily an isomorphic (isometric) transformation; hence, the preservation of distances between points is not required. Most importantly, in accordance with geometric terminology, the preservation of the relative 'distances' (in the sense of the inverse of partial order) between the points is ensured, for instance, if $d(x, z) \leq d(x, y)$, then it implies $s(x, z) \geq s(x, y)$ for any subsets $x, y, z \in \mathcal{X}$.

Example. *To illustrate with an explicit function, consider* $f(d(x, y)) = \mathrm{e}^{-\lambda d(x,y)}$ *for some* $\lambda > 0$ *and* $x, y \in \mathcal{X}$. *This exponential function is strictly decreasing. Therefore, it reverses the order of distances, establishing a similarity* s *that reflects the reversed order of the metric* d.

Furthermore, it is important to recognize that many similarity measures do not inherently relate to distance. Conversely, in numerous instances, distance metrics are derived from similarity measures. A prime example of this is the transition from Jaccard similarity to Jaccard distance [91]. This transformation highlights the necessity for the existence

of an inverse function, f^{-1}, to ensure that the conversion between similarity and distance can be both applied and reversed effectively.

2.6 Embeddings into Similarity Space

The exploration of embeddings from similarity spaces into various mathematical constructs provides a foundational framework for understanding and applying similarity spaces in diverse contexts. This section outlines the embeddings of measure space, probability space, and Hilbert space. Each embedding highlights the versatility and depth of similarity space concepts when applied to different mathematical frameworks, offering insights into the fundamental nature of similarity and its implications across various domains.

2.6.1 Embedding of Measure Space

Definition 15 (Measure Space)**.** *A measurable space is a set $\mathcal{X}$ and σ-algebra $\mathcal{S}$ of subsets of $\mathcal{X}$. A measure is an extended real valued, non-negative, and countably additive set function μ, defined on a σ-algebra $\mathcal{S}$, and such that $\mu(0) = 0$. An ordered triple $(\mathcal{X}, \mathcal{S}, \mu)$ is called a measure space.*

The meaning of this definition lies in the abstraction of measurement on countable set given by cardinality or on Lebesgue measurable set. For more details, the readers are referred to the sources [109], [110].

The measure μ of the symmetric difference of two sets can be considered as a distance between sets, well known as the *distance of Fréchet–Nikodym–Aronszajn*. This distance is a particular case of the distance in the space of Lebesgue integrable functions. In fact, the distance between sets may be treated as the distance between the characteristic functions χ_x and χ_y. These characteristic functions are defined on a set $\mathcal{X}$ and indicate membership of an element in the subset x, respectively y. In classical set theory, its value is 1 for all elements of x and 0 for all elements of $\mathcal{X}$ not in x. By employing fuzzy set theory, an uncertainty to the membership in the range of real values $\chi \in [0, 1]$ can be given.

Theorem 11 (Distance between Two Objects)**.** *Let x, y be subsets of set $\mathcal{X}$. The symmetric difference between two objects is a distance metric.*

$$d(x, y) = \mu(x \triangle y) = \int |\chi_x - \chi_y| d\mu, \tag{2.47}$$

where $x \triangle y = (x \cup y) \setminus (x \cap y)$ is the symmetric difference.

Proof. It must be shown that a distance equals the symmetric difference of two sets, expressed as $d(x, y) = \mu(x \triangle y)$ [109], [112], [113].

If μ is a σ-finite measure on a σ-algebra $\mathcal{S}$, this function is pseudometric on $\mathcal{S}$ (D1 and D2 must be satisfied), assuming $x, y, z \in \mathcal{S}$

$$\begin{aligned} d(x, z) &= \mu(x \triangle z) = \mu(z \triangle x) = \mu((x \triangle y) \triangle (y \triangle z)) \\ &\leq \mu((x \triangle y) \cup (y \triangle z)) \\ &\leq \mu(x \triangle y) + \mu(y \triangle z) \\ &= d(x, y) + d(y, z) \end{aligned} \tag{2.48}$$

The relation (D3) $x \sim y \iff d(x, y) = 0$ is an equivalence relation on $\mathcal{S}$, so d becomes a metric on the set $\mathcal{S}$. Sequential continuity with a Cauchy sequence also needs to be proven, i.e., $\{x_n\}_{n \in \mathbb{N}_0}$,

$$\lim_{n \to \infty} d(x_n, x) = 0 \implies \lim_{n \to \infty} |\mu(x_n) - \mu(x)| = 0 \tag{2.49}$$

This implies that

$$\begin{aligned} d(x, y) &= |\mu(x) - \mu(y)| = |(\mu(x \setminus y) + \mu(x \cap y)) - (\mu(x \cap y) + \mu(y \setminus x))| \\ &= |\mu(x \setminus y) - \mu(y \setminus x)| \\ &\leq |\mu(x \setminus y)| + |\mu(y \setminus x)| = \mu(x \setminus y) + \mu(y \setminus x) = \mu(x \setminus y) \cup \mu(y \setminus x) \\ &= \mu(x \triangle y) = \int |\chi_x - \chi_y| d\mu \end{aligned} \tag{2.50}$$

The symmetric difference metric is called d. The symmetric difference between two sets can be considered a measure of how 'far apart' they are. □

For better illustration, let suppose two Lebesgue measurable sets A, B. Let imagine that these sets are described by non-negative real-valued functions f, g in Cartesian system $\mathbb{R}^1$ or $\mathbb{R}^2$ (A corresponds to f and B corresponds to g) [114].

The shaded gray area at the top of Figure 2.2 essentially shows the distance between objects. Then, the area between functions f and g that corresponds area between sets A and B can be calculated. The areas in the regions can be computed $d(A, B) = \mu(A \triangle B) = \int \int |f(x, y) - g(x, y)| dx dy$ and the functions as $d(f(x), g(x)) = \mu(f \triangle g) = \int |f(x) - g(x)| dx$.

Conversely, the shaded gray area at the bottom Figure 2.2 within overlapping regions A and B and under both graphs f and g represent the similarity between those objects. Analogously, a calculation for similarity can be deduced as $s(A, B) = \mu(A \cap B) = \int \int \min\{f(x, y), g(x, y)\} dx dy$ and $s(f(x), g(x)) = \mu(f \cap g) = \int \min\{f(x), g(x)\} dx$.

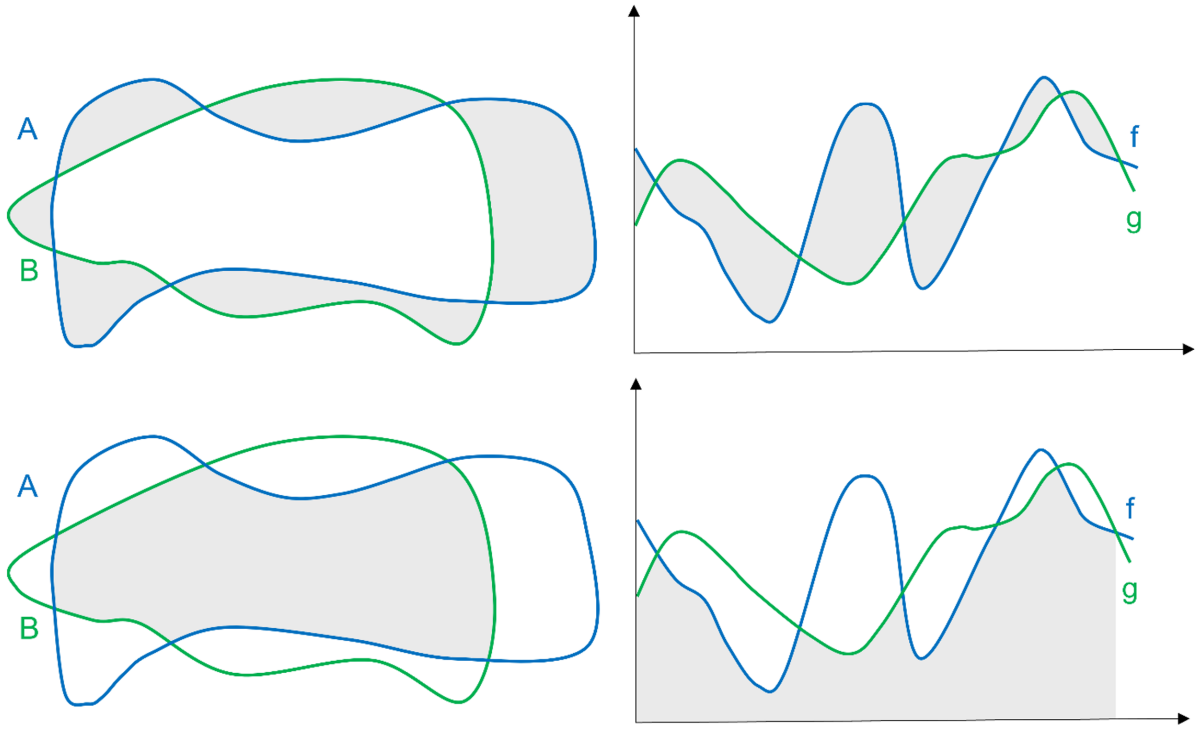

Figure 2.2: Top—symmetric difference (gray area); bottom—intersection (gray area); left—sets A and B; right—sub-graphs of f and g (inspired by [114]).

This fundamental observation allows to create a bridge between set theory and topology, such as the theories of metric spaces and similarity spaces. From the definition of the similarity $s(x, y)$ it can be deduced that the number of features shared between two objects x and y is given by their intersection $\mu(x \cap y)$. The idea behind definition is very simple, direct and intuitive too, assuming that a similarity is a measure $s(x, y) = \mu(x \cap y)$.

Theorem 12 (Similarity of Two Objects)**.** *The intersection of two objects represented by subsets x and y is a similarity*

$$s(x,y) = \mu(x \cap y) = \int \min\{\chi_x, \chi_y\} d\mu = \frac{\mu(x) + \mu(y) - \mu(x \triangle y)}{2} = \frac{\mu(x) + \mu(y) - d(x,y)}{2}, \tag{2.51}$$

Proof. Initially, the proof will focus on the relation for the intersection of the two objects.

$$\begin{aligned} s(x,y) = \mu(x \cap y) &= \mu(x) + \mu(y) - \mu(x \cup y) = \frac{2(\mu(x) + \mu(y) - \mu(x \cup y))}{2} \\ &= \frac{\mu(x) + \mu(x) + \mu(y) + \mu(y) - (\mu(x) + \mu(y) - \mu(x \cap y)) - \mu(x \cup y)}{2} \\ &= \frac{\mu(x) + \mu(y) + \mu(x \cap y) - \mu(x \cup y)}{2} = \frac{\mu(x) + \mu(y) - \mu(x \triangle y)}{2} \end{aligned} \tag{2.52}$$

The conditions S1, S3, S4 and S5 are considered trivial. The focus will be exclusively

on demonstrating S2. Since $y \supseteq (x \cap y) \cup (z \cap y)$, it follows that

$$\mu(y) \geq \mu(x \cap y) + \mu(z \cap y) - \mu(x \cap z \cap y), \tag{2.53}$$

and, consequently,

$$\mu(x \cap z) + \mu(y) \geq \mu(x \cap z \cap y) + \mu(y) \geq \mu(x \cap y) + \mu(z \cap y). \tag{2.54}$$

This yields the desired triangle inequality. □

Now the knowledge can be generalized using the similarity axioms.

Corollary 1 (Similarity of Two Objects using Duality)**.** *The similarity of two objects given by subsets $x, y \in \mathcal{X}$ is expressed*

$$s(x, y) = \frac{s(x, x) + s(y, y) - d(x, y)}{2}, \tag{2.55}$$

Proof. The self-similarity could be derived from Theorem 12

$$s(x, x) = \frac{\mu(x) + \mu(x) - d(x, x)}{2} = \frac{2\mu(x)}{2} = \mu(x) \tag{2.56}$$

Similarly, it is obtained $s(y, y) = \mu(y)$. these terms are then substituted into

$$s(x, y) = \frac{\mu(x) + \mu(y) - d(x, y)}{2} = \frac{s(x, x) + s(y, y) - d(x, y)}{2} \tag{2.57}$$

□

As a result from the proof, self-similarity is equivalent to a measure on set $\mu(x)$, e.g., cardinality of a countable set, $s(x, x) = |x|$, respectively $s(y, y) = |y|$. It is also possible to revert to the distance metric from the similarity.

Corollary 2 (Distance between Two Objects using Duality)**.** *The distance metric applied to two objects defined by subsets $x, y \in \mathcal{X}$ is given by*

$$d(x, y) = s(x, x) + s(y, y) - 2s(x, y), \tag{2.58}$$

Proof. Expressing $d(x, y)$ from Corollary 1. □

Corollary 3 (Total Dissimilarity using Duality)**.** *The total dissimilarity between two objects is given*

$$s(x, y) = \mu(x \cap y) = 0 \iff \mu(x \triangle y) = \mu(x) + \mu(y) \iff d(x, y) = s(x, x) + s(y, y), \tag{2.59}$$

Proof. Let x, y be disjoint subsets of a set $\mathcal{X}$. Let be total dissimilarity given by expression $s(x, y) = \mu(x \cap y) = 0$, thus satisfying

$$\begin{aligned} d(x,y) &= \mu(x \triangle y) = \mu((x \setminus y) \cup (y \setminus x)) = \mu(x \cup y) = \mu(x) + \mu(y) \\ &= s(x,x) + s(y,y) \end{aligned} \tag{2.60}$$

□

Total dissimilarity should mean that there are no features shared between the two objects. In set theory, this is equivalent to being a pair of disjoint sets.

Corollary 4 (Duality of Axiomatic Systems)**.** *Consider a similarity space $(\mathcal{X}, s)$ and a metric space $(\mathcal{X}, d)$. A similarity s on $\mathcal{X}$, dual to the metric d, and vice versa, a distance metric d on $\mathcal{X}$, dual to the similarity s, can be defined as follows:*

$$\begin{aligned} s(x,y) &= f \circ d(x,y) = \frac{s(x,x) + s(y,y) - d(x,y)}{2} \text{ by Corollary 1} \\ d(x,y) &= f^{-1} \circ s(x,y) = s(x,x) + s(y,y) - 2s(x,y) \text{ by Corollary 2,} \end{aligned} \tag{2.61}$$

Proof. Proceed to show the duality in this case $d(x, y) = f^{-1} \circ s(x, y) = s(x, x) + s(y, y) - 2s(x, y)$ by applying Corollary 2

$D1 \xrightarrow{f^{-1}} S1$

$$\begin{aligned} &d(x,y) = d(y,x) \\ &s(x,x) + s(y,y) - 2s(x,y) = s(x,x) + s(y,y) - 2s(y,x) \\ &s(x,y) = s(y,x) \end{aligned} \tag{2.62}$$

$D2 \xrightarrow{f^{-1}} S2$

$$\begin{aligned} &d(x,z) \leq d(x,y) + d(y,z) \\ &s(x,x) + s(z,z) - 2s(x,z) \leq s(x,x) + s(y,y) - 2s(x,y) + s(y,y) + s(z,z) - 2s(y,z) \\ &- 2s(x,z) \leq -2s(x,y) + 2s(y,y) - 2s(y,z) \\ &s(x,z) + s(y,y) \geq s(x,y) + s(y,z) \end{aligned} \tag{2.63}$$

thus, S2 is derived from Definition 5, and the triangle inequality is proven.

$D3 \xrightarrow{f^{-1}} S3$

$$
\begin{aligned}
&d(x,y) = 0 \Longrightarrow x = y \\
&s(x,x) + s(y,y) - 2s(x,y) = 0 \Longrightarrow x = y \\
&s(x,y) = s(y,y) = s(x,x) \Longrightarrow x = y
\end{aligned}
\tag{2.64}
$$

$D3 \xrightarrow{f^{-1}} S4$
$D3 \xrightarrow{f^{-1}} S5$

Since $d(x,y) = 0 \Longleftrightarrow x = y$ is bounded by zero at the same axiom, S4 and S5 are necessitated, as previously explained for Definition 5.

Similarly, the opposite approach is taken, and Corollary 1 is applied to transform $s(x,y) = f \circ d(x,y)$. □

Comparing the similarity axiom system with the partial metrics from Definition 2, the relation $p(x,y) = f^{-1} \circ d(x,y) = s(x,x) + s(y,y) - 2s(x,y)$ is observed, depending on Corollary 2 and Corollary 4, which differs from the source [106].

2.6.2 Embedding of Probability Space

The formal framework presented herein establishes the foundation for interpreting normalized similarity within the context of probability spaces. Specifically, the normalized similarity measure s_n is considered as analogous to a probability measure when applied to the intersection of events, thereby integrating the concept of similarity spaces with probability theory.

Definition 16 (Similarity Induced by Probability)**.** *Consider a probability space $(\Omega, \mathcal{F}, P)$, with Ω as the sample space, $\mathcal{F}$ as the σ-algebra of events, and P as the probability measure. For a non-empty subset $\mathcal{X} \subseteq \mathcal{F}$, two similarity measures are defined, an unnormalized similarity measure $s_u : \mathcal{X} \times \mathcal{X} \to [0,1]$ and a normalized similarity measure $s_n : \mathcal{X} \times \mathcal{X} \to [0,1]$, with the mappings defined as:*

$$
s_u(x,y) = P(x \cap y), \quad \forall x, y \in \mathcal{X}. \tag{2.65}
$$

$$
s_n(x,y) = \frac{P(x \cap y)}{P(x \cup y)}, \quad \forall x, y \in \mathcal{X}. \tag{2.66}
$$

Both s_u and s_n share compatibility with the fundamental axioms of probability, with specific properties outlined as follows:

- **Non-negativity:** $s_u(x,y), s_n(x,y) \geq 0, \forall x, y \in \mathcal{X}$

- **Triangle Inequality:**
  - For s_u: $s_u(x,z) + s_u(y,y) \geq s_u(x,y) + s_u(y,z), \forall x,y,z \in \mathcal{X}$.
  - For s_n: $s_n(x,z) + 1 \geq s_n(x,y) + s_n(y,z), \forall x,y,z \in \mathcal{X}$.
- **Identity of Indiscernibles:**
  - For s_u: Two events $x, y \in \mathcal{X}$ are considered equivalent, denoted by $x \sim_u y$, if and only if $s_u(x,y) = P(x) = P(y)$, signifying maximal self-similarity and indistinguishability in the unnormalized similarity space. The equivalence class of an event x under s_u is then given by the set:

$$[x]_{\sim_u} = \{y \in \mathcal{X} \mid s_u(x,y) = P(x) = P(y)\}, \tag{2.67}$$

    indicating that all events in $[x]_{\sim_u}$ are indistinguishable from x in terms of unnormalized similarity.
  - For s_n: Two events $x, y \in \mathcal{X}$ are equivalent, denoted by $x \sim_n y$, if and only if $s_n(x,y) = 1$. The equivalence class of an event x under s_n is defined as:

$$[x]_{\sim_n} = \{y \in \mathcal{X} \mid s_n(x,y) = 1\}, \tag{2.68}$$

    meaning that all members of $[x]_{\sim_n}$ share perfect self-similarity with x, rendering them indistinguishable within the normalized similarity space.
- **Bounded by Self-Similarity and Normalization:**
  - For s_u: $s_u(x,x) = P(x), \forall x \in \mathcal{X}$. Note that $0 \leq P(x) \leq 1$.
  - For s_n: $s_n(x,x) = 1, \forall x \in \mathcal{X}$.

Theorem 13 (Embedded Triangle Inequality for Similarity Measures)**.** *For any $x, y, z \in \mathcal{X}$, the following triangle inequalities are satisfied:*

- *For the normalized similarity measure $s_n(x,y) = \frac{P(x \cap y)}{P(x \cup y)}$:*

$$\frac{P(x \cap z)}{P(x \cup z)} + 1 \geq \frac{P(x \cap y)}{P(x \cup y)} + \frac{P(y \cap z)}{P(y \cup z)}. \tag{2.69}$$

- *For the unnormalized similarity measure $s_u(x,y) = P(x \cap y)$:*

$$P(x \cap z) + P(y) \geq P(x \cap y) + P(y \cap z). \tag{2.70}$$

Proof. The proof begins by applying the inclusion-exclusion principle to the given scenario:

$$P((x \cap y) \cup (y \cap z)) = P(x \cap y) + P(y \cap z) - P((x \cap y) \cap (y \cap z)) \leq P(y). \tag{2.71}$$

It is recognized that $(x \cap y) \cup (y \cap z)$ is a subset of $x \cup z$, leading to the conclusion:

$$P(x \cap y) + P(y \cap z) \leq P(x \cap z) + P(y). \tag{2.72}$$

For the normalized similarity measure, see [91]. □

Example (Partial Co-Occurrence $P < 1$)**.** *Considering the words "Paint" and "Point" for comparison, the analysis employs tuples of (index, character) to encapsulate both the letter and its position within the word.*

- ***Similarity Space:*** *The overlap of index-character tuples between "Paint" and "Point" is quantified, acknowledging that four out of five tuples match. This indicates a high degree of similarity, albeit not perfect due to one mismatched tuple.*
- ***Probability Space:*** *The probability measure of the intersection relative to the union of all unique index-character tuples from "Paint" and "Point" reflects a high degree of partial co-occurrence.*

In summary, the embedding of normalized similarity measures into probability spaces provides a rigorous basis for analyzing similarity in terms of probabilistic measures, specifically through the event intersections. This integration not only enriches the mathematical understanding of similarity but also expands the applicability of similarity measures in fields that rely on probabilistic reasoning.

2.6.3 Embedding of Hilbert Space

Definition 17 (Norm [7], [8], [93])**.** *Let V be a vector space over a field $\mathbb{F}$, which can be either the field of real numbers $\mathbb{R}$ or the field of complex numbers $\mathbb{C}$. A norm on V is defined as a function $\|\cdot\| : V \to \mathbb{R}_{\geq 0}$ that assigns to each vector $x \in V$ a non-negative real number $\|x\|$, satisfying the following properties for all vectors $x, y \in V$ and all scalars $\alpha \in \mathbb{F}$:*

1. ***Non-negativity:*** *$\|x\| \geq 0$, and $\|x\| = 0$ if and only if x is the zero vector, denoted as 0.*
2. ***Scalar multiplication (absolute scalability):*** *$\|\alpha x\| = |\alpha| \cdot \|x\|$*

3. ***Triangle inequality:*** $\|x + y\| \leq \|x\| + \|y\|$

The extension of the norm concept to function spaces is captured by the supremum norm (or uniform norm), denoted as $\|f\|_\infty$. For a function f defined on a domain D, the supremum norm is defined as:

$$\|f\|_\infty = \sup_{x \in D} |f(x)|, \tag{2.73}$$

where sup represents the supremum, or the least upper bound, of the set of values $|f(x)|$ for $x \in D$ [7]. This norm quantifies the maximal absolute value attained by the function f across its domain, offering a uniform measure of the function's magnitude.

The triangle inequality, a fundamental property of norms, holds in the context of function spaces and is illustrated through the supremum norm. For two functions f and g defined on the same domain D, the triangle inequality is expressed as:

$$\begin{aligned} \|f + g\|_\infty &= \sup_{x \in D} |f(x) + g(x)| && (2.74)\\ &\leq \sup_{x \in D} (|f(x)| + |g(x)|) && (2.75)\\ &\leq \sup_{x \in D} |f(x)| + \sup_{x \in D} |g(x)| = \|f\|_\infty + \|g\|_\infty. && (2.76) \end{aligned}$$

This rigorous demonstration not only validates the triangle inequality within function spaces but also reinforces the foundational principles of norm theory in a broader mathematical context.

Lemma 5 (Triangle Inequality of Norm-Induced Metrics)**.** *Given a normed space $(V, \|\cdot\|)$, the function $d : V \times V \to \mathbb{R}_{\geq 0}$ defined by $d(x, y) = \|x - y\|$ inherently satisfies the triangle inequality, which is a fundamental property for metrics. Specifically, for any $x, y, z \in V$, the triangle inequality is given by:*

$$d(x, z) \leq d(x, y) + d(y, z). \tag{2.77}$$

Proof. Given vectors $x, y, z \in V$, consider the metric defined by $d(x, y) = \|x - y\|$. To show the triangle inequality for this metric, we start by expressing $d(x, z)$ in terms of the norm: $d(x, z) = \|x - z\|$. By adding and subtracting y inside the norm, we rewrite this as $d(x, z) = \|x - y + y - z\|$. Utilizing the triangle inequality for norms, which is one of the norm axioms, we obtain $\|x - y + y - z\| \leq \|x - y\| + \|y - z\|$. By the definition of the metric, this inequality can be rewritten as $d(x, z) \leq d(x, y) + d(y, z)$.

Therefore, we have shown that the metric $d(x, y) = \|x - y\|$ induced by a norm satisfies the triangle inequality. □

Definition 18 (S-Norm for Non-Negative Functions)**.** *Let F be a set of all non-negative real-valued functions defined on a domain D. An S-Norm on F is defined as a function $[\cdot] : F \to \mathbb{R}_{\geq 0}$ that assigns to each function $f \in F$ a non-negative real number $[f]$, satisfying the following properties for all functions $f, g \in F$ and all real non-negative scalars α:*

1. ***Non-negativity:*** *$[f] \geq 0$, and $[f] = 0$ if and only if f is the zero function, which is defined as $f(x) = 0$ for all $x \in D$.*

2. ***Scalar multiplication (absolute scalability):*** *$[\alpha \cdot f] = \alpha \cdot [f]$, where $(\alpha \cdot f)(x) = \alpha \cdot f(x)$ for all $x \in D$.*

3. ***Reverse triangle inequality:*** *For the operation $+$ defined as $(f+g)(x) = f(x)+g(x)$ for all $x \in D$, it holds that $[f + g] \geq [f] + [g]$.*

Contrary to the supremum norm, the s-norm uses the infimum. The s-norm is denoted as $[f]_\infty$. For a non-negative function f defined on a domain D, the infimum s-norm is defined as:

$$[f]_\infty = \inf_{x \in D} f(x), \tag{2.78}$$

where inf represents the infimum, or the greatest lower bound, of the set of values $f(x)$ for $x \in D$.

In the domain of functional analysis, the exploration of function spaces via the s-norm explains a novel perspective on s-norm properties and their applications in functional analysis. Given two non-negative functions f and g, defined on a shared domain D, the s-norm introduces an interesting modification to the triangle inequality, as delineated below:

$$[f + g]_\infty = \inf_{x \in D}(f(x) + g(x)) \geq \inf_{x \in D} f(x) + \inf_{x \in D} g(x) \tag{2.79}$$

$$= [f]_\infty + [g]_\infty. \tag{2.80}$$

By extending classical norm space to include reverse triangle inequalities, it enriches functional analysis, shows new possibilities for theoretical exploration and practical application. The concept of the s-norm closely resembles that of the antinorm [115]–[118]. However, by applying a transformation f, this new s-norm is established, demonstrating compatibility with the duality in metric spaces not previously described in the existing concept of antinorm [OR-1].

Theorem 14 (Embedding of Hilbert Space into a Similarity Space)**.** *Let $\mathcal{X}$ be a measure space, and consider f and g as simple functions that are square-integrable in $L^2(\mathcal{X})$.*

Define a similarity space $(\mathcal{H}, s)$, where $\mathcal{H}$ is the completion of the space of simple functions in $L^2(\mathcal{X})$. The similarity function $s(f, g)$ is defined as:

$$s(f, g) = |\langle f, g \rangle|. \tag{2.81}$$

This function s quantifies the similarity between f and g in the completed space $\mathcal{H}$.

Proof. Let $\mathcal{X}$ be a measure space, and consider two simple functions f and g defined on $\mathcal{X}$. These functions can be expressed as finite sums of weighted indicator functions [93]:

$$f(x) = \sum_{i=1}^{n} a_i \chi_{A_i}(x), \tag{2.82}$$

$$g(x) = \sum_{j=1}^{m} b_j \chi_{B_j}(x), \tag{2.83}$$

where $\{A_i\}_{i=1}^n$ and $\{B_j\}_{j=1}^m$ are collections of disjoint measurable subsets of $\mathcal{X}$, a_i and b_j are real numbers, and χ_{A_i} and χ_{B_j} are the indicator functions for these sets, respectively.

The pointwise product h of f and g is defined for all $x \in \mathcal{X}$ by:

$$h(x) = f(x)g(x) = \left(\sum_{i=1}^{n} a_i \chi_{A_i}(x) \right) \left(\sum_{j=1}^{m} b_j \chi_{B_j}(x) \right), \tag{2.84}$$

which simplifies to:

$$h(x) = \sum_{i=1}^{n} \sum_{j=1}^{m} a_i b_j \chi_{A_i}(x) \chi_{B_j}(x) = \sum_{i=1}^{n} \sum_{j=1}^{m} (a_i b_j) \chi_{A_i \cap B_j}(x). \tag{2.85}$$

Here, the expression $\chi_{A_i}(x)\chi_{B_j}(x)$ signifies the indicator function of the intersection $A_i \cap B_j$, thus representing the pointwise product h as a simple function.

The inner product in L^2 space of f and g is then defined as:

$$\langle f, g \rangle = \int_{\mathcal{X}} f(x)g(x)\, d\mu = \int_{\mathcal{X}} \sum_{i=1}^{n} \sum_{j=1}^{m} a_i b_j \chi_{A_i \cap B_j}(x)\, d\mu. \tag{2.86}$$

Consequently, the integral of the pointwise product $h(x)$ equates to the inner product $\langle f, g \rangle$:

$$\langle f, g \rangle = \int_{\mathcal{X}} h(x)\, d\mu. \tag{2.87}$$

This formulation establishes a rigorous equivalence between the pointwise product of simple functions and their inner product within the context of L^2 spaces through Lebesgue integration. It also subtly suggests that the measure of intersection, $\chi_{A_i \cap B_j}(x)$, can reflect

a similarity function $s(A_i, B_j)$, further bridging the connection between measure theory and similarity measures in Hilbert spaces $(\mathcal{H}, s)$.

A similarity space $(\mathcal{H}, s)$ is complete if and only if the metric space $(\mathcal{H}, d^s)$ is complete according Lemma 4 and detailed in [119]. The space of simple functions in L^2 is not inherently complete. Completion of this space necessitates the inclusion of all limits of Cauchy sequences of simple functions, thus extending the space to encapsulate all L^2 functions [119]. □

2.7 Class of Functions Belonging to $\mathcal{C}(A)$

It is intended to be demonstrated that a substantial class of functions satisfies the axioms of similarity space, which can be formally expressed as follows:

Definition 19. *Let $\mathcal{A} \subset \mathcal{X} \times \mathcal{X}$ be a nonempty open set. The similarity is a function $s\colon \mathcal{A} \to \mathbb{R}_{\geq 0}$. It is said that s belongs to the class $\mathcal{C}$ on $\mathcal{A}$. The set of all these functions is denoted by $\mathcal{C}(\mathcal{A})$.*

From the aforementioned definition, it is sought to be shown that a large class of functions belongs to $\mathcal{C}(\mathcal{A})$. This area has not been extensively explored to date, and there are no established conventions to generalize many such similarity functions. These functions often receive different names (indices, coefficients) without consideration of their common properties and their belonging to the similarity space $(\mathcal{X}, s)$. Notice that there are many more functions belonging to $\mathcal{C}(\mathcal{A})$ that have not been presented here and will be the subject of subsequent research. The Generalized Rozinek Similarity plays a main role, as many similarity functions can be generalized by this function.

Theorem 15 (Jaccard Similarity)**.** *Jaccard similarity is a normalized similarity*

$$J_S(x, y) = \frac{\mu(x \cap y)}{\mu(x \cup y)}, \tag{2.88}$$

Proof. N1. Trivial.

N2. Since it is known that $s_n(x, y) = 1 - d_n(x, y)$, theorem 3 of [91] is modified in the dual form of Jaccard similarity instead of Jaccard distance. Then, for all sets $x, y, z \in \mathcal{X}$, from Definition 5 one has

$$J_S(x, z) + 1 \geq J_S(x, y) + J_S(y, z) \tag{2.89}$$

Let f be a nonnegative, monotone, modular set function on $\mathcal{X}$. Say that a set $\mathcal{X}$ is a null set if $f(x) = 0$. Observe that if at least one of the sets is a null set, then the

inequality is satisfied. So, it is enough to show the equivalent inequality

$$\frac{f(x \cap z)}{f(x \cup z)} + 1 \geq \frac{f(x \cap y)}{f(x \cup y)} + \frac{f(y \cap z)}{f(y \cup z)} \tag{2.90}$$

for arbitrary non-null sets $x, y, z \subseteq \mathcal{X}$. For more details of the proof, the readers are referred to [91].

N3. If $x = y \Longleftrightarrow \mu(x \cap y) = \mu(x \cup y) \Longleftrightarrow J_S(x, y) = 1$.

N4. Let x^d be any disjoint set to $\mathcal{X}$. Then $\mu(x \cap x^d) = \emptyset \Longleftrightarrow J_S(x, y) = 0$. □

The Jaccard similarity is a fundamental similarity measure on sets. Whenever it is used, it is called mainly an index or a coefficient, but it is never called a proper similarity. Note that the nonexistence of a mathematical foundation on similarity imposes the necessity of transforming the Jaccard index into the Jaccard distance $J_D(x, y) = 1 - J_S(x, y)$ and then verifying the triangle inequality $J_D(x, z) \leq J_D(x, y) + J_D(y, z)$ for that distance [91].

Theorem 16 (Jaccard Distance)**.** *The Jaccard distance is a normalized distance metric*

$$J_D(x, y) = 1 - \frac{\mu(x \cap y)}{\mu(x \cup y)} = \frac{\mu(x \triangle y)}{\mu(x \cup y)} \tag{2.91}$$

Proof. [91]. □

Theorem 17 (Rozinek Similarity [OR-1])**.** *Rozinek similarity is a normalized similarity*

$$R(x, y) = \frac{\mu(x) + \mu(y) - \mu(x \triangle y)}{\mu(x) + \mu(y) + \mu(x \triangle y)} = \frac{\mu(x) + \mu(y) - d(x, y)}{\mu(x) + \mu(y) + d(x, y)}, \tag{2.92}$$

Proof. According to the proof of Theorem 15 for $J_S(x, y)$, the equivalence with the Jaccard similarity is demonstrated as follows.

$$\begin{aligned} J_S(x, y) &= \frac{\mu(x \cap y)}{\mu(x \cup y)} = \frac{2\mu(x \cap y)}{2(\mu(x) + \mu(y) - \mu(x \cap y))} \\ &= \frac{\mu(x) + \mu(y) - \mu(x) - \mu(y) + \mu(x \cap y) + \mu(x \cap y)}{\mu(x) + \mu(y) + \mu(x) + \mu(y) - \mu(x \cap y) - \mu(x \cap y)} \\ &= \frac{\mu(x) + \mu(y) - \mu(x \cup y) + \mu(x \cap y)}{\mu(x) + \mu(y) + \mu(x \cup y) - \mu(x \cap y)} \\ &= \frac{\mu(x) + \mu(y) - \mu((x \cup y) - (x \cap y))}{\mu(x) + \mu(y) + \mu((x \cup y) - (x \cap y))} \\ &= \frac{\mu(x) + \mu(y) - \mu(x \triangle y)}{\mu(x) + \mu(y) + \mu(x \triangle y)} = R(x, y) \end{aligned} \tag{2.93}$$

□

Theorem 18 (Generalized Rozinek Similarity [OR-1])**.** *Generalized Rozinek similarity is a normalized similarity*

$$R_{GS}(x,y) = \frac{s(x,x) + s(y,y) - d(x,y)}{s(x,x) + s(y,y) + d(x,y)}, \tag{2.94}$$

Proof. Initially, the proof progresses in demonstrating $R(x,y)$ being a normalized similarity in the proof of Theorem 17. Subsequently, the obtained result is substituted into the equation of Theorem 17

$$\begin{aligned} R(x,y) &= \frac{\mu(x) + \mu(y) - d(x,y)}{\mu(x) + \mu(y) + d(x,y)} \\ &= \frac{\frac{\mu(x)+\mu(x)-d(x,x)}{2} + \frac{\mu(y)+\mu(y)-d(y,y)}{2} - d(x,y)}{\frac{\mu(x)+\mu(x)-d(x,x)}{2} + \frac{\mu(y)+\mu(y)-d(y,y)}{2} + d(x,y)} \\ &= \frac{s(x,x) + s(y,y) - d(x,y)}{s(x,x) + s(y,y) + d(x,y)} = R_{GS}(x,y). \end{aligned} \tag{2.95}$$

□

As has been proved, this similarity forms the bridge between Jaccard similarity (see Theorem 15) and similarity derived from distances. From this equation one can deduce that $\mu(x \cup y) = \frac{s(x,x)+s(y,y)+d(x,y)}{2}$.

Returning from a normalized similarity to a normalized similarity distance can be achieved by applying the inverse function.

Theorem 19 (Generalized Rozinek Normalized Distance [OR-1])**.** *Generalized Rozinek normalized distance is the following normalized distance metric*

$$R_{GD_n} = \frac{2d(x,y)}{s(x,x) + s(y,y) + d(x,y)}, \tag{2.96}$$

Proof. A direct relationship between the Jaccard distance (Theorem 16) and the generalized Rozinek normalized distance is expressed

$$\begin{aligned} J_D(x,y) &= \frac{\mu(x \triangle y)}{\mu(x \cup y)} = \frac{d(x,y)}{\mu(x \cup y)} = \frac{d(x,y)}{\frac{\mu(x)+\mu(y)+d(x,y)}{2}} \\ &= \frac{2d(x,y)}{\mu(x) + \mu(y) + d(x,y)} = \frac{2d(x,y)}{s(x,x) + s(y,y) + d(x,y)} \\ &= R_{GD_n}(x,y). \end{aligned} \tag{2.97}$$

Obviously, conditions D1 and D3 are satisfied. Reference to [91] is made for D2. □

Theorem 20 (Generalized Rozinek distance)**.** *Generalized Rozinek distance is the dis-*

tance metric

$$R_{GD}(x,y) = \frac{s(x,x) + s(y,y) - s(x,x)s(x,y) - s(y,y)s(x,y)}{s(x,y)+1}, \tag{2.98}$$

Proof. From Theorem 18, $d(x,y)$ can be expressed as follows

$$\begin{aligned}
&R_{GS}(x,y) = s(x,y) = \frac{s(x,x) + s(y,y) - d(x,y)}{s(x,x) + s(y,y) + d(x,y)} \\
&\implies s(x,y)(s(x,x) + s(y,y) + d(x,y)) = s(x,x) + s(y,y) - d(x,y) \\
&\implies s(x,x)s(x,y) + s(y,y)s(x,y) + d(x,y)s(x,y) + d(x,y) = s(x,x) + s(y,y) \\
&\implies d(x,y)(s(x,y)+1) = s(x,x) + s(y,y) - s(x,x)s(x,y) - s(y,y)s(x,y) \\
&\implies d(x,y) = \frac{s(x,x) + s(y,y) - s(x,x)s(x,y) - s(y,y)s(x,y)}{s(x,y)+1} = R_{GD}
\end{aligned} \tag{2.99}$$

From the previously proven theorems, it is obvious that $d(x,y) = \mu(x \triangle y)$ satisfies the axioms for being a distance metric. □

Theorem 21 (Tanimoto Coefficient)**.** *The Tanimoto coefficient is a generalized Rozinek similarity*

$$R_{GS}(x,y) = S(x,y) = \frac{s(x,y)}{s(x,x) + s(y,y) - s(x,y)}, \tag{2.100}$$

Proof. Upon substitution of $s(x,y)$ into the Tanimoto coefficient, it is obtained that

$$\begin{aligned}
S(x,y) &= \frac{s(x,y)}{s(x,x) + s(y,y) - s(x,y)} = \frac{\frac{s(x,x)+s(y,y)-d(x,y)}{2}}{s(x,x) + s(y,y) - (\frac{s(x,x)+s(y,y)-d(x,y)}{2})} \\
&= \frac{s(x,x) + s(y,y) - d(x,y)}{2} \frac{2}{s(x,x) + s(y,y) + d(x,y)} \\
&= R_{GS}(x,y)
\end{aligned} \tag{2.101}$$

□

Theorem 22 (Steinhaus Distance [120])**.** *Steinhaus distance is a generalized Rozinek normalized distance*

$$R_{GS}(x,y) = \sigma_\mu(f,g) = \frac{\int |f(x) - g(x)| d\mu(x)}{\int \max\{|f(x)|, |g(x)|, |f(x) - g(x)|\} d\mu(x)}, \tag{2.102}$$

Proof. Let A, B be Lebesgue measurable sets. Then, it can be written as [120]

$$\begin{aligned}
\mu(A \triangle B) &= \int |\chi_A(x) - \chi_B(x)| d\mu(x) = \int |f(x) - g(x)| d\mu(x) \\
\mu(A \cup B) &= \int \max\{\chi_A(x), \chi_B(x)\} d\mu(x) = \int \max\{|f(x)|, |g(x)|, |f(x) - g(x)|\}
\end{aligned} \tag{2.103}$$

By substituting $\mu(A \triangle B)$ and $\mu(A \cup B)$, it is obtained

$$\begin{aligned} J_D(x,y) &= \frac{\mu(A \triangle B)}{\mu(A \cup B)} = \frac{\int |f(x) - g(x)| d\mu(x)}{\int \max\{|f(x)|, |g(x)|, |f(x) - g(x)|\}} \\ &= \frac{2d(x,y)}{s(x,x) + s(y,y) + d(x,y)} \end{aligned} \tag{2.104}$$

□

Theorem 23 (Ruzicka Similarity)**.** *Ruzicka similarity [5] (generalized Jaccard similarity [121]) is a generalized Rozinek similarity*

$$R_{GS}(x,y) = J_G(x,y) = \frac{\sum_k \min\{x_k, y_k\}}{\sum_k \max\{x_k, y_k\}}, \tag{2.105}$$

Proof. For the proof, the characteric functions χ_x, χ_y and non-negative real valued functions $f(x), g(x)$ are applied:

$$\begin{aligned} J_S(x,y) &= \frac{\mu(x \cap y)}{\mu(x \cup y)} = \frac{\int \min\{\chi_x, \chi_y\} d\mu(x)}{\int \max\{\chi_x, \chi_y\} d\mu(x)} = \frac{\int \min\{f(x), g(x)\} d\mu(x)}{\int \max\{f(x), g(x)\} d\mu(x)} \\ &= \lim_{\delta x \to 0} \frac{\sum \min\{f(x), g(x)\} \delta x}{\sum \max\{f(x), g(x)\} \delta x} = J_G(x,y) \end{aligned} \tag{2.106}$$

In the last step, the continous functions are discretized, where δ may be chosen as a sampling length. The relation of $J_S(x,y)$ to $R_{GS}(x,y)$ is derived in Appendix 18. □

The distance derived from the Ruzicka similarity $d_n(x,y) = 1 - J_G(x,y)$ is known by the name 'Soergel distance' [5].

Theorem 24 (Gaussian Similarity)**.** *Gaussian similarity is a similarity*

$$s(x,y) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{\frac{-(x-y)^2}{2\sigma^2}\right\} \approx \exp\{-(x-y)^2\} = \exp\{-d(x,y)^2\}, \tag{2.107}$$

Proof. The properties S1, S3, S4 and S5 are trivial. S2 is satisfied as follows

$$\begin{aligned} &\big(1 - \exp\{-d(x,y)^2\}\big)\big(1 - \exp\{-d(y,z)^2\}\big) \geq 0 \\ &\exp\{-d(x,y)^2 - d(y,z)^2\} + 1 \geq \exp\{-d(x,y)^2\} + \exp\{-d(y,z)^2\} \\ &\exp\{-d(x,z)^2\} + 1 \geq \exp\{-d(x,y)^2\} + \exp\{-d(y,z)^2\} \\ &s(x,z) + s(y,y) \geq s(x,y) + s(y,z) \end{aligned} \tag{2.108}$$

□

Gaussian similarity is relevant to the natural human and animal perception of similarity based on psychological research [14], where it is shown that a stimulus decays

exponentially with the distance. Numerous experiments have provided empirical observations of learned responses to some measure of different stimuli. As the independent variable of a physical measure of the difference between two stimuli, there have been chosen, for example, the difference in wavelengths of light, frequencies of tones or angular orientations of shapes.

Theorem 25 (Rozinek Natural Distance)**.** *The Rozinek natural distance is a distance metric*

$$R_{ND}(x,y) = \sigma\sqrt{-2\ln\left(\sigma\sqrt{2\pi}s(x,y)\right)} \approx \sqrt{-\ln(s(x,y))}, \tag{2.109}$$

Proof. Expressing $d(x,y)$ from Theorem 24. □

This distance is derived from Gaussian similarity and describes an inverse problem of how human and animal perception treats a distance depending on the similarity. In addition, there is a limit $\lim_{s(x,y)\to 0^+} R_{ND}(x,y) = +\infty$.

In cases where the similarity measurement is only dependent on the distance and Jaccard-like similarities cannot be used directly, for example, for an edit distance—also called the *k difference problem* [122]—the similarity is very appropriate. An analogy between the k difference problem and the symmetric difference set $x \triangle y$ in set theory can be observed.

Theorem 26 (Normalized Edit Similarity)**.** *The normalized edit similarity is a Rozinek similarity over the alphabet* Σ

$$s_n(x,y) = \frac{|x| + |y| - d(x,y)}{|x| + |y| + d(x,y)}, \tag{2.110}$$

where $d(x,y)$ *is an edit distance.*

Proof. **Case 1: Levenshtein similarity**

$$\begin{aligned} R(x,y) &= \frac{\mu(x) + \mu(y) - d(x,y)}{\mu(x) + \mu(y) + d(x,y)} = \frac{|x| + |y| - d(x,y)}{|x| + |y| + d(x,y)} = 1 - \frac{2d(x,y)}{|x| + |y| + d(x,y)} \\ &= 1 - d_{N-GLD}(x,y) \end{aligned} \tag{2.111}$$

where $d_{N-GLD}(x,y)$ is a normalized generalized Levenshtein distance of the form

$$d_{N-GLD}(x,y) = \frac{2d(x,y)}{\alpha(|x| + |y|) + d(x,y)} \tag{2.112}$$

where $\alpha = 1$ is the minimum cost of insertion and deletion costs and $d(x,y)$ is an edit distance [123]. This proof has been inspired by [2], [111] where it is further proved that d_{N-GLD} is a normalized distance metric. Hence by the duality between normalized similarity and normalized distance metrics, $s_n(x,y) = 1 - d_n(x,y)$ is proven.

Case 2: Longest Common Subsequence (LCS)

The same results are obtained when normalizing the LCS. Let l be the length of the LCS [124]

$$l(x,y) = \frac{1}{2}(|x| + |y| - d_{LCS}(x,y)) \tag{2.113}$$

where l(x,y) satisfies the similarity axioms from Definition 5 and d_{LCS} denotes the edit distance based on unit insertion and deletation cost [2]. Now attention is turned to normalizing similarity through evaluating a generalized Tanimoto coefficient [2], [90]

$$S(x,y) = \frac{s(x,y)}{s(x,x) + s(y,y) - s(x,y)}. \tag{2.114}$$

The $s(x,y)$ is interpreted as a count of common features, while $S(x,y)$ express this count as a fraction of the total number of features of x and y. Setting $s(x,y) = l(x,y)$, it is thus obtained

$$S(x,y) = \frac{l(x,y)}{|x| + |y| - l(x,y)} \tag{2.115}$$

Since $l(x,x) = |x|$ and $l(x,y) = |y|$, the above expressions are elaborated to

$$S(x,y) = \frac{|x| + |y| - d_{OM}(x,y)}{|x| + |y| + d_{OM}(x,y)} \tag{2.116}$$

where d_{OM} is an edit distance (for details see [2]). Thus, it is proved that also $S(x,y) = R(x,y)$. □

The normalized edit similarity is suitable for conversion from Levenshtein distance or the normalization of the longest common subsequence (LCS). The procedure is shown in proofs.

Chapter 3

Linear Regression in Similarity Space

In the vast realm of data analysis, the treatment and understanding of textual data, particularly database records, have become increasingly paramount. As the digital universe grows exponentially, with an estimated 2.5 quintillion bytes of data produced daily, a significant portion of this avalanche is textual data [125]. These are the records that detail transactions, logs, communications, and countless other human and machine interactions. Understanding the patterns, structures, and relationships within this textual data offers profound opportunities for knowledge discovery and decision support [126].

Traditional data analysis methodologies, prominently the least squares regression, have been foundational in the domain of quantitative data [127]. Rooted in the early 19th century and attributed to Legendre [128] and Gauss [129], the least squares method has been an invaluable tool in deducing relationships within data, finding applications from astronomy to economics. But can this time-tested method be adapted to the nuanced realm of textual data?

Textual data, unlike quantitative data, primarily relies on the notion of 'similarity' rather than 'magnitude' [88]. Two words or phrases may not exhibit a quantifiable difference, but they can show varying degrees of similarity based on their semantics, usage, or context [130]. Because of this property of textual data, there's a need for an analysis method that recognizes its qualitative aspect. This has led to the idea of modifying the least squares method to work within a similarity framework.

In this thesis, the application of least squares regression in the context of textual data similarity is investigated. A method is introduced to adapt traditional regression techniques to work in a similarity space, with a primary focus on word similarities in database records. The approach involves defining an appropriate similarity, reshaping the problem space, and ensuring results are both robust and interpretable [131].

With this research, the intention is to enhance data analysis techniques, bridging the gap between the quantitative precision of traditional methods and the qualitative depth

of textual data [126].

The primary areas of focus in this thesis are regression analysis and similarity spaces, aiming to adapt the regression analysis framework to function within similarity spaces.

3.1 Problem Formulation

Linear regression is a fundamental statistical and machine learning technique used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. The formulation of a linear regression problem involves several key components, which can be described as follows:

Definition 20 (Linear Regression [OR-3], [132]–[134])**.** *Given a dataset* $\{y_i, x_{i,1}, ..., x_{i,m}\}_{i=1}^{n}$ *consisting of* n *observations, each observation includes a dependent (target) variable* y_i *and* m *independent (predictor) variables* $\mathbf{x}_i = [1, x_{i,1}, x_{i,2}, \ldots, x_{i,m}]^T$*. The goal of linear regression is to find a linear relationship between the dependent variable and the independent variables.*

The linear relationship is represented by the equation:

$$y_i = \theta_0 + \theta_1 x_{i,1} + \ldots + \theta_m x_{i,m} = \mathbf{x}_i^T \boldsymbol{\theta} + \epsilon_i \tag{3.1}$$

where the coefficients are denoted by $\boldsymbol{\theta} = [\theta_0, \theta_1, \ldots, \theta_m]^T$ *including* θ_0 *as the intercept term, and* ϵ_i *is the error term for the* i*-th observation, indicating the differences between observed and predicted values of the dependent variable.*

The objective of linear regression is to find the values of the coefficients $\boldsymbol{\theta}$ that minimize the differences between the predicted and observed values of the dependent variable. This is often achieved by minimizing the cost function, which is typically the Mean Squared Error (MSE) for all observations [128], [129], [132], [133]:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J_{\text{MSE}}(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{3.2}$$

where $\hat{y}_i$ is the predicted value of the dependent variable for the i-th observation, calculated as:

$$\hat{y}_i = \mathbf{x}_i^T \boldsymbol{\theta}. \tag{3.3}$$

The function $J(\boldsymbol{\theta})$ represents the cost associated with a particular set of parameters $\boldsymbol{\theta}$, and the goal is to find the set of $\boldsymbol{\theta}$ that minimizes $J(\boldsymbol{\theta})$. Although MSE is a common choice for the cost function in linear regression, other cost functions can also be used depending on the specific requirements and characteristics of the data.

In this thesis, instead of adhering to the traditional view, a dualistic perspective within a similarity space is introduced and explained through the following definition.

Definition 21 (Objective Function in Similarity Space [OR-3], [OR-4])**.** *Let V be a vector space over a field $\mathbb{F}$. Consider a similarity space (V, s) with a function $s\colon V \times V \to \mathbb{R}_{\geq 0}$. Formally, the objective function of linear regression in this context aims at maximizing the similarity over the similarity space (V, s), which can be formally expressed as*

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} s(y_i, \hat{y}_i) \tag{3.4}$$

for any $y_i, \hat{y}_i \in V$, where y_i is the dependent (target) variable and $\hat{y}_i$ is the predicted value of the dependent variable for the i-th observation.

3.2 State-of-the-Art

In the field of computer science, linear regression remains a fundamental technique for modeling the relationship between variables. Over the years, various objective functions and modifications have been proposed to enhance the performance and versatility of linear regression models. In this section, an overview of the state-of-the-art objective functions and their popular modifications is provided.

The most widely used objective function for linear regression is Ordinary Least Squares (OLS). OLS aims to minimize the sum of squared differences between predicted and actual target values [128], [129].

3.2.1 Regularized Regression

In the pursuit of addressing overfitting and enhancing model generalization, regularized linear regression methods have gained substantial popularity. These methods augment the traditional OLS objective function by incorporating penalty terms that encourage specific properties in the model. Here, three widely used regularized linear regression techniques are discussed:

Ridge Regression (L2 Regularization)

Ridge regression, introduced by Tikhonov [135], extends the OLS framework by adding an L2 regularization term to the objective function. This regularization term penalizes the magnitude of the coefficients, thereby encouraging smaller coefficients. The Ridge Regression objective function is defined as:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J_{\mathrm{R}}(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 + \lambda \sum_{j=1}^{n} \theta_j^2 \tag{3.5}$$

Here, $\boldsymbol{\theta}$ denotes the model parameters (coefficients), $\hat{y}_i$ is the predicted value, y_i is the actual target value, n is the number of features, and λ is the regularization parameter.

Ridge regression encourages smaller coefficient values, effectively reducing overfitting and improving model generalization.

Lasso Regression (L1 Regularization)

Lasso regression, introduced by Tibshirani [136], promotes sparsity within the model by incorporating an L1 penalty term in the objective function. The Lasso Regression objective function is defined as:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J_{\mathrm{L}}(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 + \lambda \sum_{j=1}^{n} |\theta_j| \tag{3.6}$$

Similar to Ridge regression, the model parameters θ are adjusted during training, but Lasso encourages some coefficients to be exactly zero, effectively performing feature selection.

Elastic Net Regression (L1 + L2 Regularization)

Elastic Net Regression, proposed by Zou and Hastie [137], strikes a balance between Ridge and Lasso regression by combining both L1 and L2 regularization terms. The Elastic Net objective function is defined as:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J_{\mathrm{EN}}(\theta) \tag{3.7}$$

$$= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 + \lambda_1 \sum_{j=1}^{n} |\theta_j| + \lambda_2 \sum_{j=1}^{n} \theta_j^2 \tag{3.8}$$

Here, λ_1 and λ_2 are regularization parameters that control the strength of L1 and L2 regularization, respectively.

Elastic Net provides a versatile regularization approach, allowing users to balance feature selection and regularization according to their specific needs.

3.2.2 Robust Regression

In scenarios where the data may be contaminated with outliers, robust regression techniques have been developed to provide more resilient modeling. These methods aim to minimize the impact of outliers on the model while still capturing the underlying trends in the majority of the data. One notable approach is Huber loss, introduced by Huber [138].

Huber Loss

Huber loss combines the benefits of both mean squared error (MSE) and mean absolute error (MAE) and is designed to handle data with outliers more effectively. The Huber Loss objective function is defined as follows:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J_{\mathrm{H}}(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} L_{\delta}(\hat{y}_i - y_i) \tag{3.9}$$

In this equation, n represents the number of data points, $\boldsymbol{\theta}$ denotes the model parameters (coefficients), $\hat{y}_i$ is the predicted value, and y_i is the actual target value. The function L_{δ} is a piecewise loss function that combines the properties of MSE for small errors and MAE for large errors:

$$L_{\delta}(z) = \begin{cases} \frac{1}{2}z^2, & \text{if } |z| \leq \delta \\ \delta(|z| - \frac{1}{2}\delta), & \text{if } |z| > \delta \end{cases}$$

Here, δ is a tuning parameter that controls the threshold for switching between the quadratic and linear regions. For small errors ($|z| \leq \delta$), the loss is quadratic, similar to MSE. For large errors ($|z| > \delta$), the loss is linear, similar to MAE.

Huber loss offers a robust alternative to traditional OLS by providing a balanced approach to handle outliers while still maintaining the benefits of squared loss for small errors. It is widely used in regression tasks where data quality and robustness to outliers are critical considerations.

3.3 Simple Linear Regression in Similarity Space

A vector space equipped with an inner product $|\langle \cdot, \cdot \rangle|$ is termed a Hilbert space if it is complete with respect to the norm induced by that inner product, i.e., $\|x\| = \sqrt{\langle x, x \rangle}$. Under certain conditions, if similarity is measured based on the Lebesgue measure, the

equation

$$s(x, y) = \mu(x \cap y) = |\langle x, y \rangle| \tag{3.10}$$

holds, where μ denotes the Lebesgue measure. This measure satisfies certain axioms as given in Definition 5 and proved in Theorem 14.

Given a dataset with observed values y and input values $\mathbf{x}_i$, the goal is to find parameters that maximize the projection of y onto its predicted values $\hat{y} = \theta_0 + \theta_1 x$. This can be quantified using the dot product of these vectors:

$$\hat{\boldsymbol{\theta}} = [\hat{\theta_0}, \hat{\theta_1}]^T = \arg\max_{\theta_0, \theta_1} \left\{ \sum_{i=1}^{n} y_i \hat{y}_i - \lambda \sum_{i=1}^{n} \hat{y_i}^2 \right\} \tag{3.11}$$

$$= \arg\max_{\theta_0, \theta_1} \left\{ \sum_{i=1}^{n} y_i (\theta_0 + \theta_1 x_i) - \lambda \sum_{i=1}^{n} (\theta_0 + \theta_1 x_i)^2 \right\} \tag{3.12}$$

where λ is a regularization coefficient that controls the balance between maximizing the projection and minimizing the magnitude of the parameters.

3.3.1 Deriving the Gradients

To find the values of θ_0 and θ_1 that maximize the objective function, the gradient is computed and set to zero.

For θ_0:

$$\frac{\partial J}{\partial \theta_0} = \sum_{i=1}^{n} y_i - 2\lambda \sum_{i=1}^{n} (\theta_0 + \theta_1 x_i) \tag{3.13}$$

$$= n\overline{y} - 2\lambda(n\overline{\theta_0} + \overline{\theta_1 x}) \tag{3.14}$$

where $\overline{y}$ is the mean of the observed values, and $\overline{\theta_0}$ and $\overline{\theta_1 x}$ are the means of the predicted values.

For θ_1:

$$\frac{\partial J}{\partial \theta_1} = \sum_{i=1}^{n} y_i x_i - 2\lambda \sum_{i=1}^{n} x_i (\theta_0 + \theta_1 x_i) \tag{3.15}$$

$$= \sum_{i=1}^{n} y_i x_i - 2\lambda (\sum_{i=1}^{n} x_i \theta_0 + \sum_{i=1}^{n} \theta_1 x_i^2) \tag{3.16}$$

Setting these gradients to zero yields the conditions for the optimal parameters θ_0 and θ_1. The gradient ascent updates for the parameters at each iteration are given by:

$$\theta_0 \leftarrow \theta_0 + \alpha \frac{\partial J}{\partial \theta_0} \tag{3.17}$$

$$\theta_1 \leftarrow \theta_1 + \alpha \frac{\partial J}{\partial \theta_1} \tag{3.18}$$

where α is the learning rate, which controls the size of the steps taken in the direction of the gradient. The optimization continues until a maximum number of iterations is reached or the change in the objective function value between successive iterations is less than a specified tolerance ϵ. The detailed steps are presented in Algorithm 1 [OR-3], [OR-4].

Robustness to Outliers

Consider two datasets: one without outliers D and one with an outlier D'. Let the objective values for these datasets be represented as $J(D)$ and $J(D')$.

Without the regularization term, the difference in objectives due to an outlier is:

$$\Delta J_{\text{no-reg}} = J(D') - J(D) \tag{3.19}$$

$$= \sum_{i \in D'} y_i(\theta_0 + \theta_1 x_i) - \sum_{i \in D} y_i(\theta_0 + \theta_1 x_i) \tag{3.20}$$

Given the influence of an outlier, this difference could be significantly large.

However, with the regularization term:

$$\Delta J_{\text{reg}} = J(D') - J(D) \tag{3.21}$$

$$= \left[\sum_{i \in D'} y_i(\theta_0 + \theta_1 x_i) - \lambda \sum_{i \in D'} (\theta_0 + \theta_1 x_i)^2 \right] \tag{3.22}$$

$$- \left[\sum_{i \in D} y_i(\theta_0 + \theta_1 x_i) - \lambda \sum_{i \in D} (\theta_0 + \theta_1 x_i)^2 \right] \tag{3.23}$$

The regularization term, $-\lambda \sum_i (\theta_0 + \theta_1 x_i)^2$, penalizes large values of the parameters, thereby limiting the magnitude of predictions.

To examine the impact of an outlier on this regularized objective, consider a single outlier point $(x_{\text{out}}, y_{\text{out}})$ such that y_{out} is much larger than other values.

The contribution of this outlier to the objective is:

$$\Delta J_{\text{outlier}} = y_{\text{out}}(\theta_0 + \theta_1 x_{\text{out}}) - \lambda(\theta_0 + \theta_1 x_{\text{out}})^2 \tag{3.24}$$

While the data term $y_{\text{out}}(\theta_0 + \theta_1 x_{\text{out}})$ tries to fit the outlier closely, the regularization

term $-\lambda(\theta_0 + \theta_1 x_{\text{out}})^2$ prevents the model parameters from adapting too much to the outlier. Thus, a suitable choice of λ can limit the outlier's influence, ensuring robustness.

Convergence to Linear Regression Model

To demonstrate that the regularized objective function converges towards a linear regression solution, the properties of the objective function and the gradient ascent update rules need to be examined. Given that the function is maximized, it needs to be concave. This function is a dot product minus a sum of squared parameters (a regularization term). Under certain conditions, this function can be concave, especially if the dot product term dominates the behavior. The negative sum of squares (regularization term) is concave. For concave functions, gradient ascent with a suitable learning rate guarantees convergence to a global maximum. And the potential concavity of the objective function, these updates will iteratively increase the value of J until it converges to its global maximum, fitting the model to the given data.

The regularization term penalizes extreme values of $\hat{\boldsymbol{\theta}}$. This ensures that the algorithm doesn't diverge, promoting stability and convergence. Moreover, the regularization can be viewed as a form of penalty that keeps the parameter values bounded, ensuring that the gradient ascent does not lead to unbounded growth of the parameters.

Chapter 4

Fixed-Point Theory in Similarity Space

4.1 Problem Formulation

Similarity and dissimilarity functions are essential tools in numerous research fields, such as information retrieval, data mining, machine learning, cluster analysis, and various applications in database search and protein sequence comparison. The use of dissimilarity functions typically necessitates a metric space, which is a well-defined mathematical structure. However, the concept of similarity functions lacks a formally accepted definition, leading to ambiguity and inconsistency in their utilization.

To address this gap, the aim is to establish a viable theory of similarity space by constructing it as a duality to metric space. This approach allows for a more rigorous mathematical understanding of similarity functions and their properties, providing a foundation for further research and applications.

Fixed point theory is a central topic in modern mathematics and non-linear analysis, providing a powerful and versatile tool for addressing various problems across diverse fields. Researchers often formulate problems in terms of finding fixed points of specific mappings when studying the solvability of functional equations. Fixed point theory has broad applications in biology, chemistry, economics, game theory, optimization theory, and physics [139], with numerous esteemed mathematicians, such as Cauchy, Fredholm, Liouville, Lipschitz, Peano, Picard, and Nash, contributing to its development.

Given the importance of fixed point theory, investigating its potential within similarity spaces is an essential step towards better understanding the properties and applications of similarity functions. Such research can lead to the development of new theoretical frameworks, which can, in turn, contribute to improvements in various mathematical techniques and models. By examining fixed points in similarity spaces, insights into the

connections between similarity and metric spaces can be gained, as well as the potential applicability of fixed point theorems in this novel contex

A typical fixed point problem can be formulated as follows: Let $\mathcal{X}$ be a given set, and let be a pair of non-empty sets $M, S \in \mathcal{X}$ such that $M \cap S \neq \emptyset$. Given a mapping $T\colon M \to S$, the interest lies in finding a point $x \in M$ such that $Tx = x$, which is referred to as a fixed point of T [139]–[142]. This raises three primary questions of interest:

- *Existence:* When does problem have at least one solution?
- *Uniqueness:* If problem has a solution, when is such solution unique?
- *Approximation:* In the case of uniqueness, the question arises of how to develop a numerical algorithm that converges to the solution?

In this thesis, the questions of existence and uniqueness for a contraction principle in similarity space are addressed. By investigating this mapping within the context of fixed point theory, the aim is to deepen an understanding of similarity spaces and explore potential applications for this mathematical concept in various research areas.

Throughout this thesis, the set of all real numbers is denoted by $\mathbb{R}$, the set of positive real numbers by $\mathbb{R}_{>0} = \{x \in \mathbb{R} \mid x > 0\}$, the set of non-negative real numbers by $\mathbb{R}_{\geq 0} = \{x \in \mathbb{R} \mid x \geq 0\}$, and the set of all natural numbers by $\mathbb{N}$.

In the following text, new results and terms are introduced. The similarity space has not been studied from this perspective yet, and such results are quite new.

Similarity and dissimilarity functions are essential tools in numerous research fields, including information retrieval, data mining, machine learning, cluster analysis, and various applications in database searches and protein sequence comparisons. The use of dissimilarity functions typically necessitates a metric space, which is a well-defined mathematical structure. However, the concept of similarity functions lacks a formally accepted definition, resulting in ambiguity and inconsistency in their utilization. To address this gap, the aim is to establish a viable theory of similarity space by constructing it as a duality to metric space [OR-1].

4.2 State-of-the-Art

Historically, the distinction between the three key areas of Fixed Point Theory was established through the discovery of three major theorems:

- **Topological Fixed Point Theory:**
  - *Brouwer's Fixed Point Theorem (1912)* states that every continuous function mapping a compact convex subset of a Euclidean space to itself has at least one fixed point [143].
- **Metric Fixed Point Theory:**
  - *Banach's Fixed Point Theorem (1922)*, or the Contraction Mapping Theorem, asserts that a contraction mapping on a complete metric space must have a unique fixed point [144].
- **Discrete Fixed Point Theory:**
  - *Tarski's Fixed Point Theorem (1955)* extends the concept of Brouwer's theorem to complete lattices, demonstrating the existence of fixed points in these structures [145].

Further, the study in Fixed Point Theory also involves analysis of:

- **Single-valued Mapping:** Functions where each element in the domain maps to a single element in the target space.
- **Set-valued Mapping (Multivalued Mapping):** Functions that may assign a set of points in the target space to each domain element.

In this state-of-the-art research, the focus is on the single-valued mapping of fixed points in metric spaces, due to its close structure to similarity spaces and the fact that the theory of fixed points in this structure is being introduced for the first time. The most significant theorems are chronologically arranged to enhance understanding of the current state and development in fixed point theory. For a broader context, there are books that provide an extensive overview, such as those by Pathak (2018) [141], Subrahmanyam (2014)[140], Agarwal (2018) [139], Almazel (2014) [108], and Kirk (2014) [142].

4.2.1 Banach's Fixed Point

Banach fixed-point (1922) theorem [144], a fundamental cornerstone of metric space theory, serves as a powerful tool for a myriad of analytical problems. The theorem has been extensively studied, emphasizing its diverse applications in analysis. The Banach fixed-point theorem not only asserts the existence and uniqueness of fixed points for certain self-mappings in metric spaces, but also provides a constructive method for their discovery, thus endowing it with significant practical utility.

Theorem 27 (Banach [144]). *Let $(\mathcal{X}, d)$ be a complete metric space. Suppose $T : \mathcal{X} \to \mathcal{X}$ is a contraction mapping, i.e., there exists a constant $\alpha \in (0, 1)$ such that $d(Tx, Ty) \leq \alpha d(x, y)$ for all $x, y \in \mathcal{X}$. Then*

(i) T has a unique fixed point $x^* \in \mathcal{X}$.

(ii) For every $x \in \mathcal{X}$, the Picard sequence $\{T^n x\}_{n\in\mathbb{N}}$ converges to x^*:

$$\lim_{n\to\infty} T^n x = x^*. \tag{4.1}$$

(iii) For every $x \in \mathcal{X}$, the following speed of convergence is observed:

$$d(T^n x, x^*) \leq \frac{\alpha^n}{1-\alpha} d(x, Tx) \tag{4.2}$$

for $n \in \mathbb{N}$.

Proof. (i) The existence of the fixed point is first demonstrated. Any $x_0 \in \mathcal{X}$ is chosen, and a sequence $\{T^n x\}_{n\in\mathbb{N}}$ in $\mathcal{X}$ is defined for $n \geq 0$. It is claimed that $\{T^n x\}_{n\in\mathbb{N}}$ forms a Cauchy sequence in $\mathcal{X}$. Indeed, for $m > n$,

$$\begin{aligned} d(T^n x, T^{n+m} x) &\leq d(T^n x, T^{n+1} x) + \cdots + d(T^{n+m-1} x, T^{n+,} x) \\ &\leq (\alpha^n + \cdots + \alpha^{n+m-1}) d(x, Tx) \\ &\leq (\alpha^n + \alpha^{n+1} + \alpha^{n+2} + \cdots) d(x, Tx) \\ &\leq \frac{\alpha^n}{1-\alpha} d(x, Tx). \end{aligned} \tag{4.3}$$

Since $\alpha \in (0, 1)$, $\alpha^n \to 0$ as $n \to \infty$. Hence $\{x_n\}$ is a Cauchy sequence. As $\mathcal{X}$ is complete, $\{T^n x\}_{n\in\mathbb{N}}$ converges to an element $x^* \in \mathcal{X}$.

Now, it is shown that x^* is a fixed point of T. Since T is continuous (as a contraction mapping),

$$Tx^* = T(\lim_{n\to\infty} T^n x) = \lim_{n\to\infty} T(T^n x) = \lim_{n\to\infty} T^{n+1} x = x^*.$$

(ii) For uniqueness, suppose y^* is another fixed point of T, $y^* \neq x^* \in \mathcal{X}$ and $y^* = Ty^*$, then $d(x^*, y^*) > 0$ and

$$d(x^*, y^*) = d(Tx^*, Ty^*) \leq \alpha d(x^*, y^*) < d(x^*, y^*), \tag{4.4}$$

a contradiction. Hence $d(x^*, y^*) = 0$, that is $x^* = y^*$.

(iii) By the triangle inequality, it is established that

$$d(T^n x, u) \leq d(T^n x, T^p x) + d(T^p x, u) \leq \frac{\alpha^n}{1-\alpha} d(Tx, x) + d(T^p x, x^*) \tag{4.5}$$

for $n < p$ by the inequality. Letting $p \to \infty$, the result is obtained

$$d(T^n x, x^*) \leq \frac{\alpha^n}{1-\alpha} d(Tx, x), \tag{4.6}$$

which provides a control on the convergence rate of $\{T^n x\}_{n\in\mathbb{N}}$ to the fixed point x^*.

□

4.2.2 Rakotch's Extension

Rakotch (1962) provided a further generalization for a broader class of contractions. The Rakotch Fixed Point Theorem is formulated as:

Theorem 28 (Rakotch [146]). *Let $(\mathcal{X}, d)$ be a complete metric space and $T : \mathcal{X} \to \mathcal{X}$ a mapping. Suppose there exists a decreasing function $\varphi : \mathbb{R}_{\geq 0} \to [0, 1)$ such that for all $x, y \in \mathcal{X}$,*

$$d(Tx, Ty) \leq \varphi(d(x, y))d(x, y). \tag{4.7}$$

Then T has a fixed point.

4.2.3 Kannan's Fixed Point

Kannan (1969) introduced a different type of contraction condition. The Kannan Fixed Point Theorem is stated as:

Theorem 29 (Kannan [147]). *Let $(\mathcal{X}, d)$ be a complete metric space and $T : \mathcal{X} \to \mathcal{X}$ a continuous mapping. Suppose there exists a constant $\alpha \in [0, \frac{1}{2})$ such that for all $x, y \in \mathcal{X}$, the following contraction condition holds:*

$$d(Tx, Ty) \leq \alpha[d(x, Tx) + d(y, Ty)]. \tag{4.8}$$

Then, the mapping T has a fixed point.

4.2.4 Meir-Keeler Contractions

Shortly after Ciric's work, Meir and Keeler (1969) proposed a new type of contraction condition. Their generalization relaxed the strict requirements of earlier contraction definitions.

Theorem 30 (Meier-Keeler [148]). *Let $(\mathcal{X}, d)$ be a complete metric space and $T : \mathcal{X} \to \mathcal{X}$ a mapping. Meir and Keeler introduced the condition that for every $\varepsilon > 0$, there exists $\delta > 0$ such that for all $x, y \in \mathcal{X}$,*

$$d(x, y) < \varepsilon + \delta \implies d(Tx, Ty) < \varepsilon. \tag{4.9}$$

Under this condition, T has a fixed point.

4.2.5 Boyd-Wong Generalization

One of the main generalizations of the Banach principle is the theorem proposed by D.W. Boyd and J.S. Wong (1969) in [149].

The range of d is denoted by P, and the closure of P is denoted by $\overline{P}$, so $P = \{d(x, y) \mid x, y \in \mathcal{X}\}$. This is also applicable within the context of a similarity space.

Theorem 31 (Boyd–Wong [149]). *Let $(\mathcal{X}, d)$ be a complete metric space with P defined as $d(x, y) \colon x, y \in \mathcal{X}$. Let $T \colon X \to \mathcal{X}$ be a self-mapping satisfying:*

$$d(Tx, Ty) \leq \psi(d(x, y)) \tag{4.10}$$

for each $x, y \in \mathcal{X}$ where $\psi \colon \overline{P} \to \mathbb{R}_{\geq 0}$ is upper semicontinuous from the right on $\overline{P}$ and satisfies $\psi(t) < t$ for all $t \in \overline{P} \setminus \{0\}$, where $\overline{P}$ denotes the closure P. Then, T has a unique fixed point x^* and $d(T^n x, x^*) \to 0$ for each $x \in \mathcal{X}$.

The Boyd–Wong theorem's applicability has been extensively studied across various abstract mathematical spaces. Notably, its principles have been examined in the context of partially ordered metric spaces [150], cone metric spaces [151], and generalized metric spaces [152]. Further investigations have been carried out in partial metric spaces [105], [153], [154], quasi-metric spaces [155], b-metric spaces [156], and bipolar metric spaces [157]. Other applications and generalizations can be found in [158], [159].

4.2.6 Chatterjea's Fixed Point

Chatterjea (1972) proposed a theorem similar to Kannan's but involving the distance between points and their images under different mappings. The Chatterjea Fixed Point Theorem is given by:

Theorem 32 (Chatterjea [160]). *Let $(\mathcal{X}, d)$ be a complete metric space and $T : \mathcal{X} \to \mathcal{X}$ be a mapping such that there exists a number $\alpha \in [0, \frac{1}{2})$ such that ,*

$$d(Tx, Ty) \leq \alpha[d(x, Ty) + d(y, Tx)], \tag{4.11}$$

for all $x, y \in (\mathcal{X}$. Then, T has a unique fixed point in $\mathcal{X}$.

4.2.7 Ciric's Generalization

Ljubomir Ciric in 1974 proposed a generalization that unified several existing fixed point theorems, including the Banach and Kannan fixed point theorems.

Theorem 33 (Ciric [161])**.** *Let $(\mathcal{X}, d)$ be a complete metric space, and let $T : \mathcal{X} \to \mathcal{X}$ be a quasicontraction, that is, for a fixed constant $\alpha < 1$,*

$$d(Tx, Ty) \leq \alpha \max\{d(x, y), d(x, Tx), d(y, Ty), d(x, Ty), d(y, Tx)\}, \quad \forall x, y \in \mathcal{X}. \quad (4.12)$$

Then, T has a unique fixed point.

4.2.8 Matkowski Generalization

Matkowski (1975) replaced the condition of upper semi-continuity on ψ by Boyd–Wong condition and state and proved the following theorem.

Theorem 34 (Matkowski [162])**.** *Let $(\mathcal{X}, d)$ be a complete metric space and suppose that $T\colon \mathcal{X} \to \mathcal{X}$ satisfies*

$$d(Tx, Ty) \leq \psi(d(x, y)) \quad (4.13)$$

for all $x, y \in \mathcal{X}$, where $\psi\colon \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ is monotone non-decreasing and satisfies $\lim_{n\to\infty} \psi^n(t) = 0$ for all $t > 0$. Then T has a unique fixed point in $\mathcal{X}$.

4.2.9 Caristi-Ekeland Fixed Point

Caristi's Fixed Point Theorem, introduced in 1976, provides a novel approach to fixed point theory based on the Ekeland Variational Principle, linking it to the concept of lower semicontinuous functions.

Theorem 35 (Caristi-Ekeland [163], [164])**.** *Let $(\mathcal{X}, d)$ be a complete metric space and $\varphi : \mathcal{X} \to \mathbb{R}$ a lower semicontinuous function bounded below. If there exists a mapping $T : \mathcal{X} \to \mathcal{X}$ such that for all $x \in \mathcal{X}$,*

$$d(x, Tx) \leq \varphi(x) - \varphi(Tx), \quad (4.14)$$

then T has a fixed point.

4.2.10 Rhoades' Generalization

In 1977, Rhoades made a significant contribution by further relaxing the conditions of the Banach contraction principle. His generalization is particularly important for its applications in the analysis of non-expansive mappings.

Theorem 36 (Rhoades [165]). *Let $(\mathcal{X}, d)$ be a complete metric space, and suppose that $T\colon \mathcal{X} \to \mathcal{X}$ satisfies the following inequality:*

$$d(Tx, Ty) \leq d(x, y) - \psi(d(x, y)), \tag{4.15}$$

for all $x, y \in \mathcal{X}$, where $\psi\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a continuous and nondecreasing function such that $\psi(t) = 0$ if and only if $t = 0$. Then, f has a unique fixed point.

4.2.11 Suzuki Fixed Point

This theorem, introduced by Tomonari Suzuki in 2008, is particularly interesting because it establishes conditions under which a mapping in a complete metric space guarantees the existence of a unique fixed point, utilizing a novel approach involving a specifically defined non-increasing function.

Theorem 37 (Suzuki [166]). *Let $(\mathcal{X}, d)$ be a complete metric space, and suppose that $T : \mathcal{X} \to \mathcal{X}$. Define a non-increasing function $\psi : [0, 1) \to \left(\frac{1}{2}, 1\right]$ by*

$$\psi(\alpha) = \begin{cases} 1, & \text{if } 0 \leq \alpha \leq \frac{1}{2}\left(\sqrt{5} - 1\right), \\ \frac{1-\alpha}{\alpha^2}, & \text{if } \frac{1}{2}\left(\sqrt{5} - 1\right) < \alpha < \frac{1}{\sqrt{2}}, \\ \frac{1}{1+\alpha}, & \text{if } \frac{1}{\sqrt{2}} \leq \alpha < 1. \end{cases} \tag{4.16}$$

Assume that there exists $\alpha \in [0, 1)$ such that

$$\psi(h)d(x, Tx) \leq d(x, y) \Rightarrow d(Tx, Ty) \leq \alpha d(x, y), \text{ for all } x, y \in \mathcal{X}. \tag{4.17}$$

Then, T has a unique fixed point.

4.2.12 Wardowski Fixed Point

In 2012, Dariusz Wardowski introduced new types of fixed point theorems in metric spaces, which are based on functions that satisfy certain conditions, known as F-contractions.

Theorem 38 (Wardowski [167]). *Let $(\mathcal{X}, d)$ be a complete metric space and $T : X \to X$ a mapping. Wardowski introduced the concept of F-contractions, where a function F :*

$\mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ *satisfies certain specific conditions. If for all* $x, y \in X$,

$$F(d(Tx, Ty)) \leq F(d(x, y)) - \psi(d(x, y)), \tag{4.18}$$

where ψ *is a lower semicontinuous function from* $\mathbb{R}_{>0}$ *to* $\mathbb{R}_{>0}$, *then* T *has a fixed point.*

4.3 Similarity Contraction Principle

The objective of this section is to establish some generalized properties of the similarity space version of the Banach fixed point theorem.

The construction of the similarity space using f results in a continuous, monotonically decreasing, and mapping of $d(x, y)$. However, it is important to note that this mapping does not preserve the metric structure of the original space. Instead, it forms a similarity space. In this new similarity space, the introduction of the new definition for mappings that exhibit similar properties is required.

The presentation of the similarity contraction principle involves the introduction of the analogous concept of Lipschitz mappings in this newly formed space. Additionally, the extension of the Banach contraction principle to the similarity space setting is proposed. These concepts will play a crucial role in establishing the desired generalized properties.

Definition 22 (s-Lipschitz Mapping)**.** *Given two similarity spaces* (X, s_X) *and* (Y, s_Y) *where* s_X *denotes similarity on the non-empty set* X *and* s_Y *is the similarity on non-empty set* Y, *a function* $T\colon X \to Y$ *is called* s-Lipschitz continuous *if there exists an s-Lipschitz constant* $k > 0$ *such that for all* $x, y \in X$*:*

$$s_Y(Tx, Ty) \geq k \cdot s_X(x, y). \tag{4.19}$$

An s-Lipschitz mapping with an s-Lipschitz constant $k < 1$ *is called similarity contraction.*

Lemma 6. *Given two similarity spaces* (X, s_X) *and* (Y, s_Y), *a function* $T : X \to Y$ *is uniformly s-continuous if it is a s-Lipschitz mapping.*

Proof. Assume that $T : X \to Y$ is a s-Lipschitz mapping with s-Lipschitz constant $k > 0$. Let $\epsilon > 0$ and choose $\delta = \frac{\epsilon}{k}$. Then for all $x, y \in \mathcal{X}$ such that $s_X(x, y) > \min\{s_X(x, x), s_X(y, y)\} - \delta$, it is observed that

$$\begin{aligned} s_Y(Tx, Ty) &\geq k \cdot s_X(x, y) \\ &> k \cdot (\min\{s_X(x, x), s_X(y, y)\} - \delta) \\ &\geq \min\{s_Y(Tx, Tx), s_Y(Ty, Ty)\} - \epsilon. \end{aligned} \tag{4.20}$$

Hence, T is uniformly s-continuous on $\mathcal{X}$. □

The classes of s-Lipschitz continuous, uniformly s-continuous, and s-continuous functions are denoted by $\mathcal{L}$, $\mathcal{U}$, and $\mathcal{C}$ respectively. These classes are related as $\mathcal{L} \subsetneq \mathcal{U} \subsetneq \mathcal{C}$. This represents the fact that every s-Lipschitz continuous function is uniformly s-continuous, and every uniformly s-continuous function is s-continuous, but the converse is not necessarily true.

Definition 23 (Similarity Space Contraction)**.** *A mapping $T : X \to X$ is said to be contraction in a similarity space (X, s) if there exists a constant $\alpha \in (0, 1)$ such that similarity contraction condition*

$$s(Tx, Ty) \geq \alpha \cdot s(x, y) \tag{4.21}$$

for all $x, y \in X$ are satisfied.

The utilization of a duality in defining contraction in similarity spaces, encapsulated by the self-similarity and mutual similarity contraction conditions, marks a nuanced divergence from traditional metric space contractions. In the domain of similarity spaces, defining contraction requires a careful delineation to preserve the inherent structure and properties, particularly in relation to traditional metric spaces. The proposed definition establishes a bifurcated condition for contraction—self-similarity and mutual similarity—ensuring that both intrinsic and comparative similarity are bounded and well-regulated. The need for this dual condition emerges distinctly from the nuanced nature of similarity spaces, wherein simply satisfying intrinsic (self) similarity contraction doesn't invariably imply an overarching, consistent contraction within the entire similarity space.

Furthermore, the subsequent theorem and its proof seek to establish a robust mathematical bridge between similarity space contractions and the traditional Lipschitz condition in metric spaces, affording a streamlined transition and applicability of established metric space theories in the rich, flexible domain of similarity spaces.

Theorem 39 (Dualistic Similarity Contractive Mapping)**.** *Let $(\mathcal{X}, s)$ be a similarity space, and let $T : \mathcal{X} \to \mathcal{X}$ be mapping. The mapping T is said to be* contractive *if there is a real number $\alpha \in (0, 1)$ such that*

$$s(Tx, Tx) + s(Ty, Ty) - 2s(Tx, Ty) \leq \alpha\big[s(x, x) + s(y, y) - 2s(x, y)\big]. \tag{4.22}$$

Proof. The similarity contraction conditions defined previously are applied

$$s(Tx, Tx) \leq \alpha \cdot s(x, x) \wedge s(Tx, Ty) \geq \alpha \cdot s(x, y) \tag{4.23}$$

$$s(Ty, Ty) \leq \alpha \cdot s(y, y) \wedge s(Tx, Ty) \geq \alpha \cdot s(x, y) \tag{4.24}$$

and by summing all inequalities and their combinations, the result should be obtained

$$\begin{gathered} s(Tx, Tx) + s(Ty, Ty) - 2s(Tx, Ty) \leq \alpha(s(x, x) + s(y, y) - 2s(x, y)) \\ d^s(Tx, Ty) \leq \alpha d^s(x, y). \end{gathered} \tag{4.25}$$

A form of induced elementary metric is observed as given by Theorem 6, and finally, a Lipschitz mapping is obtained in the metric space $(\mathcal{X}, d)$ such that $d(Tx, Ty) \leq kd(x, y)$ denoting $d^s(Tx, Ty) = d(Tx, Ty)$, $d^s(x, y) = d(x, y)$ and $k = \alpha \in (0, 1)$. □

Corollary 5. *Let $(\mathcal{X}, s)$ be a similarity space, and let $T : \mathcal{X} \to \mathcal{X}$ be a mapping that satisfies the inequality*

$$s(Tx, Tx) - s(Tx, Ty) \leq \alpha(s(x, x) - s(x, y)), \tag{4.26}$$

for all $x, y \in \mathcal{X}$ and for some real number $\alpha \in (0, 1)$. Then, T is a contractive mapping with constant $\alpha \in (0, 1)$.

Proof. Given a mapping $T : \mathcal{X} \to \mathcal{X}$ that satisfies the inequalities

$$s(Tx, Tx) \leq \alpha s(x, x) \wedge s(Tx, Ty) \geq \alpha s(x, y) \tag{4.27}$$

these inequalities are subtracted, and by factoring out α, the result is obtained

$$s(Tx, Tx) - s(Tx, Ty) \leq \alpha(s(x, x) - s(x, y)). \tag{4.28}$$

which is the condition given in the statement. Hence, if T satisfies this inequality for all $x, y \in \mathcal{X}$. □

Lemma 7 (Generalized Triangle Inequality)**.** *Consider a set of elements $x_1, x_2, ..., x_n \in X$ in a similarity space $(\mathcal{X}, s)$. Then generalized triangle inequality is following:*

$$s(x_1, x_n) + \sum_{i=2}^{n-1} s(x_i, x_i) \geq \sum_{i=1}^{n-1} s(x_i, x_{i+1}). \tag{4.29}$$

Proof. Base case ($n = 3$):
The inequality needs to be shown to hold for $n = 3$.

$$s(x_1, x_3) + s(x_2, x_2) \geq s(x_1, x_2) + s(x_2, x_3) \tag{4.30}$$

This is the standard triangle inequality (S2) from Definition 5 in a similarity space, which holds true for any similarity space $(\mathcal{X}, s)$. Thus, the base case is established.

Inductive step: Assume that the inequality holds for $n = k$:

$$s(x_1, x_k) + \sum_{i=2}^{k-1} s(x_i, x_i) \geq \sum_{i=1}^{k-1} s(x_i, x_{i+1}). \tag{4.31}$$

It needs to be shown that the inequality also holds for $n = k + 1$:

$$s(x_1, x_{k+1}) + \sum_{i=2}^{k} s(x_i, x_i) \geq \sum_{i=1}^{k} s(x_i, x_{i+1}). \tag{4.32}$$

To prove this, the triangle inequality in the similarity space can be utilized:

$$s(x_1, x_{k+1}) + s(x_k, x_k) \geq s(x_1, x_k) + s(x_k, x_{k+1}). \tag{4.33}$$

If the inductive assumption (4.31) is added to inequality (4.33), the following is obtained:

$$s(x_1, x_{k+1}) + \sum_{i=2}^{k} s(x_i, x_i) \geq \sum_{i=1}^{k} s(x_i, x_{i+1}). \tag{4.34}$$

Thus, it has been shown that if the inequality holds for $n = k$, it also holds for $n = k+1$. According to the principle of mathematical induction, the generalized triangle inequality holds for any sequence of elements in a similarity space $(\mathcal{X}, s)$. □

Theorem 40 (Normalized Similarity Contraction Principle)**.** *Let $(\mathcal{X}, s_n)$ be a complete normalized similarity space. Let $T\colon \mathcal{X} \to \mathcal{X}$ be a contraction mapping, with constant $\alpha \in (0, 1)$. Then*

(i) T has a unique fixed point $x^* \in \mathcal{X}$.

(ii) For every $x \in \mathcal{X}$, the Picard sequence $\{T^n x\}_{n\in\mathbb{N}}$ converges to x^*:

$$\lim_{n\to\infty} T^n x = x^*. \tag{4.35}$$

(iii) The following speed of convergence is observed for every $x \in \mathcal{X}$:

$$s(T^n x, x^*) \geq 1 - \frac{\alpha^n}{1-\alpha}\Big(1 - s(x, Tx)\Big) \tag{4.36}$$

for $n \in \mathbb{N}$.

Proof. Claim (i)
Suppose, for the sake of contradiction, that there exist distinct points $x^*, y^* \in X$ such that $Tx^* = x^*$ and $Ty^* = y^*$. Applying the self-similarity contraction condition, it is obtained

that $s(Tx^*, Tx^*) = s(x^*, x^*) \geq \alpha s(x^*, x^*)$. Similarly, the mutual similarity contraction condition gives $s(Tx^*, Ty^*) = s(x^*, y^*) \geq \alpha s(x^*, y^*)$. Summing these inequalities yields:

$$s(x^*, y^*) + s(x^*, x^*) \geq \alpha s(x^*, y^*) + \alpha s(x^*, x^*). \tag{4.37}$$

Including the non-negativity condition, it is obtained that

$$s(x^*, y^*) \geq \frac{\alpha - 1}{1 - \alpha} s(x^*, x^*) \ngeq 0. \tag{4.38}$$

This is a contradiction for $\alpha \in (0, 1)$. Therefore, the conclusion is reached that if $Tx = x^*$ and $Ty^* = y^*$, then it must be $s(x^*, x^*) = s(y^*, y^*)$, leading to $x^* = y^*$. Thus, the fixed point of T is unique.

Claim (ii) and (iii)

Let $x \in \mathcal{X}$ be an arbitrary point. Utilizing the fact that T is a similarity contraction mapping with constant α, and employing the self-similarity contraction condition and the mutual similarity contraction condition, the following inequalities can be derived:

$$d^s(Tx, Ty) \leq \alpha d^s(x, y) \tag{4.39}$$

$$1 - s(Tx, Ty) \leq \alpha - \alpha s(x, y) \tag{4.40}$$

$$-s(Tx, Ty) \leq -1 + \alpha - \alpha s(x, y) \tag{4.41}$$

$$s(Tx, Ty) \geq 1 - \alpha + \alpha s(x, y) \tag{4.42}$$

It is aimed to be proved by induction that for $n \geq 1$,

$$s(T^n x, T^n y) \geq 1 - \alpha^n + \alpha^n s(x, y). \tag{4.43}$$

For $n = 2$, applying T to both sides of the inequality yields:

$$s(T^2 x, T^2 y) \geq 1 - \alpha + \alpha s(Tx, Ty). \tag{4.44}$$

Substituting the inequality $s(Tx, Ty) \geq 1 - \alpha + \alpha s(x, y)$ into the right-hand side:

$$s(T^2 x, T^2 y) \geq 1 - \alpha + \alpha(1 - \alpha + \alpha s(x, y)). \tag{4.45}$$

Expanding and simplifying:

$$s(T^2 x, T^2 y) \geq 1 - \alpha^2 + \alpha^2 s(x, y). \tag{4.46}$$

Assume the statement is true for some $k \geq 2$, i.e.,

$$s(T^k x, T^k y) \geq 1 - \alpha^k + \alpha^k s(x, y). \tag{4.47}$$

It must be shown that it holds for $k + 1$:

$$\begin{aligned} s(T^{k+1}x, T^{k+1}y) &\geq 1 - \alpha + \alpha s(T^k x, T^k y), \\ &\geq 1 - \alpha + \alpha(1 - \alpha^k + \alpha^k s(x, y)), \\ &\geq 1 - \alpha^{k+1} + \alpha^{k+1} s(x, y). \end{aligned}$$

This completes the inductive step, showing the pattern holds for all n.

By induction, it has been established that for any $n \geq 1$,

$$s(T^n x, T^n y) \geq 1 - \alpha^n + \alpha^n s(x, y). \tag{4.48}$$

This shows the similarity measure s between any two points transformed by T^n is bounded from below by a function of α^n and the original similarity measure $s(x, y)$, highlighting how the contraction property of T influences the similarity measure over iterations.

Using generalized triangle inequality according to Lemma 7, for $(n, m) \in \mathbb{N} \times \mathbb{N} \setminus \{0\}$, the result is obtained

$$\begin{aligned} &s(T^n x, T^{n+m} x) + s(T^{n+1} x, T^{n+1} x) + ... + (T^{n+m-1} x, T^{n+m-1} x) \\ &\geq s(T^n x, T^{n+1} x) + ... + s(T^{n+m-1} x, T^{n+m} x). \end{aligned} \tag{4.49}$$

By shortening the notation in sums, it is obtained that

$$\begin{aligned} s(T^n x, T^{n+m}x) &\geq \sum_{i=n}^{n+m-1} s(T^i x, T^{i+1}x) - \sum_{i=n+1}^{n+m-1} s(T^i x, T^i x) \\ &= \sum_{i=n}^{n+m-1} s(T^i x, T^{i+1}x) - (m-2) \\ &\geq m - 1 - \sum_{i=n}^{n+m-1} \alpha^i + \sum_{i=n}^{n+m-1} \alpha^i s(x, Tx) - (m-2) \\ &= 1 + \sum_{i=n}^{n+m-1} \alpha^i s(x, Tx) - \sum_{i=n}^{n+m-1} \alpha^i \\ &= 1 + \sum_{i=n}^{n+m-1} \alpha^i \Big(s(x, Tx) - 1 \Big) \\ &= 1 + \alpha^n \frac{1-\alpha^m}{1-\alpha} \Big(s(x, Tx) - 1 \Big) \\ &\geq 1 + \frac{\alpha^n}{1-\alpha} \Big(s(x, Tx) - 1 \Big) \\ &= 1 - \frac{\alpha^n}{1-\alpha} \Big(1 - s(x, Tx) \Big) \end{aligned} \tag{4.50}$$

Therefore, given that an arbitrary $k > n$, it is observed that

$$s(T^n x, T^k x) \geq 1 - \frac{\alpha^n}{1-\alpha} \Big(1 - s(x, Tx) \Big). \tag{4.51}$$

It follows that $\lim_{n,k\to\infty} s(T^n x, T^k x) \to 1$ as $n \to \infty$. Thus for all $k \in \mathbb{N}$ the induced elementary metric in similarity space,

$$\begin{aligned} &\lim_{n,k\to\infty} d^s(T^n x, T^k x) \\ &= \lim_{n\to\infty} s(T^n x, T^n x) + \lim_{k\to\infty} s(T^k x, T^k x) - \lim_{n,k\to\infty} 2s(T^n x, T^k x) \\ &= 0. \end{aligned} \tag{4.52}$$

Therefore $\{T^n x\}_{n\in\mathbb{N}}$ is a Cauchy sequence in $(\mathcal{X}, d^s)$. Since $(\mathcal{X}, s)$ is complete, so is $(\mathcal{X}, d^s)$ and the sequence $\{T^n x\}_{n\in\mathbb{N}}$ converges to some $x^* \in X$ with respect to the metric d^s. Let $\epsilon > 0$ be arbitrarily fixed. It is established that $\lim_{n\to\infty} \alpha^n = 0$ and $\lim_{k\to\infty} \alpha^k = 0$.

Therefore

$$\begin{aligned} s(x^*, x^*) &= \lim_{n\to\infty} s(T^n x, x^*) = \lim_{n,k\to\infty} s(T^n x, T^k x) \\ &= \lim_{n,k\to\infty} \min\Big\{ s(T^n x, T^n x), s(T^k x, T^k x) \Big\} \\ &= \min\Big\{ \lim_{n\to\infty} s(T^n x, T^n x), \lim_{k\to\infty} s(T^k x, T^k x) \Big\} \\ &\geq \min\Big\{ \lim_{n\to\infty} (1 - \alpha^n + \alpha^n s(x^*, x^*)), \lim_{k\to\infty} (1 - \alpha^k + \alpha^k s(x^*, x^*)) \Big\} \\ &= 1 - \alpha^k (1 - s(x^*, x^*)) = 1 \\ &> 1 - \epsilon \end{aligned} \tag{4.53}$$

for all $n, k > N$, with $n \to \infty$, it is observed that $s(Tx^*, x^*) = s(x^*, x^*)$, thereby implying $x^* = Tx^*$. □

4.4 Boyd-Wong Theorems

The main contribution is the study of the dual relationship between the similarity space and the metric space, where the similarity space forms a different axiomatic system. The focus is placed on the dualistic view of the Boyd–Wong contraction, purely from the perspective of similarity spaces. The derivations are demonstrated through several examples.

Theorem 41 (Boyd-Wong Dualistic Contraction [OR-9])**.** *Let $(\mathcal{X}, s)$ be a complete similarity space and so its dual complete metric space $(\mathcal{X}, d^s)$ and let $T\colon \mathcal{X} \to \mathcal{X}$ satisfy the following condition:*

$$\begin{aligned} &s(Tx, Tx) + s(Ty, Ty) - 2s(Tx, Ty) \\ &\leq \psi(s(x, x)) + \psi(s(y, y)) - 2\psi(s(x, y)), \quad \forall x, y \in \mathcal{X} \end{aligned} \tag{4.54}$$

where $\psi\colon \overline{P} \to \mathbb{R}_{\geq 0}$ is lower semicontinuous from the right on $\overline{P}$ and $\psi(t) > t$ for all $t \in \overline{P} \setminus \{0\}$. Then, T has a unique fixed point and every sequence $\{T^n x\}_{n\in\mathbb{N}}$ converges to this unique fixed point x^*.

Proof. Let $x, y \in \mathcal{X}$, and let $\varphi\colon \overline{P} \to \mathbb{R}_{\geq 0}$ is upper semicontinuous from the right on $\overline{P}$ and $\varphi(t) < t$ for all $t \in \overline{P} \setminus \{0\}$. Given that a complete similarity space implies a complete metric space, it may be assumed that

$$d^s(x, y) \leq \varphi(d^s(x, y)). \tag{4.55}$$

Furthermore, it is expressed from the previous inequality that

$$\begin{aligned} & s(Tx,Tx)+s(Ty,Ty)-2s(Tx,Ty) \\ & \leq \varphi(s(x,x)+s(y,y)-2s(x,y)) \\ & \leq \psi(s(x,x))+\psi(s(y,y))-2\psi(s(x,y)). \end{aligned} \tag{4.56}$$

So the claim is proven. □

In similarity spaces, defining contraction involves a unique duality, represented by self-similarity and mutual similarity conditions. This stands in contrast to traditional metric space contractions. The proposed definition establishes a bifurcated condition for contraction—self-similarity and mutual similarity—ensuring that both intrinsic and comparative similarity are bounded and well-regulated.

Theorem 42 (Boyd-Wong Similarity Contraction [OR-9])**.** *Let $\mathcal{X}$ be a complete similarity space, and let $T\colon \mathcal{X} \to \mathcal{X}$ satisfy the contraction conditions of similarity*

$$s(Tx,Ty) \geq \psi(s(x,y)), \quad \forall x,y \in \mathcal{X} \tag{4.57}$$

where $\psi\colon \overline{P} \to \mathbb{R}_{\geq 0}$ is lower semicontinuous from the right on $\overline{P}$ and $\psi(t) > t$ for all $t \in \overline{P} \setminus \{0\}$. Then,

(i) T has a unique fixed point $x^* \in X$,

(ii) for every $x \in \mathcal{X}$, the Picard sequence $\{T^n x\}_{n\in\mathbb{N}}$ converges to x^*:

$$\lim_{n\to\infty} T^n x = x^*. \tag{4.58}$$

Proof. Let $x \in \mathcal{X}$, define a shorter notation for self-similarity and mutual similarity

$$u_n = s(T^n x, T^n x) \qquad \text{(self-similarity)}, \tag{4.59}$$

$$w_n = s(T^n x, T^{n-1} x) \qquad \text{(mutual similarity)}. \tag{4.60}$$

The sequences $\{u_n\}_{n\in\mathbb{N}}$ and $\{w_n\}_{n\in\mathbb{N}}$ are monotonically increasing. Since both sequences are bounded, they are convergent. Let denote the limits of these sequences as $\lim_{n\to\infty} u_n = u$ and $\lim_{n\to\infty} w_n = w$. To ensure the theorem's conditions are satisfied, it needs to be shown that $u_n, w_n \to u, w$ as $n \to \infty$. However, given that $u, w > 0$t is found that

$$\begin{aligned} u_{n+1} &\geq \psi(u_n), \\ w_{n+1} &\geq \psi(w_n). \end{aligned} \tag{4.61}$$

So that

$$\begin{aligned} u &= \lim_{n\to\infty} u_n = \liminf_{n\to\infty} u_n \geq \liminf_{t\to u_+} \psi(t) \geq \psi(u), \\ w &= \lim_{n\to\infty} w_n = \liminf_{n\to\infty} w_n \geq \liminf_{t\to w_+} \psi(t) \geq \psi(w), \end{aligned} \tag{4.62}$$

which is a contradiction because $\psi(u) \not> u$ and $\psi(w) \not> w$ as in the statement. Thus, u_n, w_n converges to similarities u, w as $n \to \infty$ for each $x \in \mathcal{X}$.

It is now shown that $\{T^n x\}_{n\in\mathbb{N}}$ is a Cauchy sequence for each $x \in \mathcal{X}$. This completes the proof, as the limit of this sequence, a fixed point x^* of T, is clearly unique. Should $\{T^n x\}_{n\in\mathbb{N}}$ not be a Cauchy sequence, then for some $\epsilon, \delta > 0$ and for each $k \in \mathbb{N}$, sequences of natural numbers $\{m(k)\}_{n\in\mathbb{N}}$ and $\{n(k)\}_{n\in\mathbb{N}}$ can be found with $m(k) > n(k) \geq k$ such that for all $k \in \mathbb{N}$, applying the closed s-ball and its complement from Definition 7 in this manner:

$$\begin{aligned} s_k = s(T^{m(k)}x, T^{n(k)}x) &\leq \delta \\ &= \max\{s(T^{m(k)}x, T^{m(k)}x), s(T^{n(k)}x, T^{n(k)}x)\} - \epsilon \end{aligned} \tag{4.63}$$

and

$$\begin{aligned} s(T^{m(k)-1}x, T^{n(k)}x) &> \delta \\ &= \max\{s(T^{m(k)-1}x, T^{m(k)-1}x), s(T^{n(k)}x, T^{n(k)}x)\} - \epsilon. \end{aligned} \tag{4.64}$$

This can be accomplished by choosing $m(k)$ as the least natural number exceeding $n(k)$ for satisfying the above inequality. Now,

$$\begin{aligned} s_k &= s(T^{m(k)}x, T^{n(k)}x) \\ &\geq s(T^{m(k)}x, T^{m(k)-1}x) + s(T^{m(k)-1}x, T^{n(k)}x) - s(T^{m(k)-1}x, T^{m(k)-1}x) \\ &\geq w_{m(k)} + \delta - u_{m(k)}. \end{aligned} \tag{4.65}$$

Thus, $\delta \geq s_k \geq w_{m(k)} + \delta - u_{m(k)}$. Consequently,

$$\begin{aligned} \delta &\geq \limsup_{k\to\infty} s_k \geq \liminf_{k\to\infty} s_k \geq \liminf_{k\to\infty}(w_{m(k)} + \delta - u_{m(k)}) \\ &= \liminf_{k\to\infty} w_{m(k)} + \delta - \limsup_{k\to\infty} u_{m(k)} \\ &= \delta. \end{aligned} \tag{4.66}$$

Hence, $\lim_{k\to\infty} s_k = \delta$. Indeed, $s_k \to \delta^+$ as $k \to \infty$.

Further,

$$\begin{aligned} s_k &= s(T^{m(k)}x, T^{n(k)}x) \\ &\geq s(T^{m(k)}x, T^{m(k)+1}x) + s(T^{m(k)+1}x, T^{n(k)+1}x) + s(T^{n(k)+1}x, T^{n(k)}x) \\ &\quad - s(T^{m(k)+1}x, T^{m(k)+1}x) - s(T^{n(k)+1}x, T^{n(k)+1}x) \\ &\geq w_{m(k)} + \psi(s(T^{m(k)}x, T^{n(k)}x) + w_{n(k)} - u_{m(k)+1} - u_{n(k)+1} \\ &\geq 2w_k + \psi(s_k) - 2u_k. \end{aligned} \tag{4.67}$$

Taking the limit as $k \to +\infty$ in the above inequality, it follows that

$$\liminf_{k\to\infty} s_k = \lim_{k\to\infty} s_k = \delta^+ \geq \liminf_{k\to\infty} \psi(s_k) \geq \psi(\delta). \tag{4.68}$$

Since $\delta > 0$, this contradicts that $\delta > \psi(\delta)$. Hence, $\{T^n x\}_{n\in\mathbb{N}}$ is a Cauchy sequence in $\mathcal{X}$. As $\mathcal{X}$ is complete, it converges to an element x^* in $\mathcal{X}$. Since for all $x,\ y \in \mathcal{X}$, $s(Tx, Ty) \geq \psi(s(x, y)$, T is continuous. Since $\{T^{n+1}x\}_{n\in\mathbb{N}}$ converges to Tx^* and is also a subsequence of $\{T^n x\}_{n\in\mathbb{N}}$, it follows that $x^* = Tx^*$. Since $\psi(t) > t$ for all $t > 0$, it follows that the fixed point x^* of T is unique. □

4.5 Applications

4.5.1 Solution to Fredholm Integral Equation

Suppose that the similarity space of s-continuous functions $\mathcal{S}([a, b], \mathbb{R})$ is complete with similarity

$$s_\infty(u, v) = \inf_{t\in[a,b]} \{|u(t)|, |v(t)|\} \tag{4.69}$$

where $u, v \in \mathcal{S}([a, b], \mathbb{R})$ are s-continuous real functions.

Theorem 43. *Let $f : [a, b] \to \mathbb{R}$ be a s-continuous function, and let $K : [a, b] \times [a, b] \times \mathbb{R} \to \mathbb{R}$ be a function such that the kernel $K(t, s, u)$ is s-continuous in all its arguments. Consider the nonlinear Fredholm integral equation of the second kind:*

$$u(t) = f(t) + \lambda \int_a^b K(t, s, u(s))\, ds, \quad t \in [a, b], \tag{4.70}$$

where λ is a given constant. Suppose the following conditions are satisfied:

(a) ***s-Continuity****: The kernel $K(t, s, u)$ is s-continuous in t, s, and u over the domain $[a, b] \times [a, b] \times \mathbb{R}$.*

(b) ***Compactness***: *The operator* $(Tu)(t) = \int_a^b K(t, s, u(s))\, ds$ *is compact on the space of s-continuous functions* $\mathcal{S}([a, b], \mathbb{R})$.

(c) ***Boundedness***: *There exists a constant* $M > 0$ *such that* $M \leq |K(t, s, u)|$ *for all* $t, s \in [a, b]$ *and for all* $u \in \mathbb{R}$.

(d) ***s-Lipschitz Condition***: *There exists a s-Lipschitz constant* $L > 0$ *such that similarity condition*

$$\inf_{t\in[a,b]} \{|K(t, s, u)|, |K(t, s, v)|\} \geq L \inf_{t\in[a,b]} \{|u(t)|, |v(t)|\}, \tag{4.71}$$

for all $t, s \in [a, b]$ *and for all* $u, v \in \mathbb{R}$.

Then, there exists at least one function $u \in \mathcal{S}([a, b], \mathbb{R})$ *that satisfies the nonlinear Fredholm integral equation of the second kind.*

Proof. Consider the nonlinear Fredholm integral equation of the second kind:

$$u(t) = f(t) + \lambda \int_a^b K(t, s, u(s))\, ds, \quad t \in [a, b], \tag{4.72}$$

where f is a s-continuous function on $[a, b]$, and $K(t, s, u)$ is s-continuous in its arguments, with $\lambda \in [0, 1)$.

Define the operator $T : \mathcal{S}([a, b], \mathbb{R}) \to \mathcal{S}([a, b], \mathbb{R})$ by

$$(Tu)(t) = f(t) + \lambda \int_a^b K(t, s, u(s))\, ds. \tag{4.73}$$

Step 1: Show that T *is a contraction.*

For T to be a contraction, there needs to be a constant $\alpha \in (0, 1)$ such that for all functions u, v in $\mathcal{S}([a, b], \mathbb{R})$, similarity contraction condition

$$\inf_{t\in[a,b]} \{|(Tu)(t)|, |(Tv)(t)|\} \geq \alpha \inf_{t\in[a,b]} \{|u(t)|, |v(t)|\} \tag{4.74}$$

for each $t \in [a, b]$.

Given the s-Lipschitz condition on K, for any $t \in [a, b]$, similarity contraction condition

and assuming $f(t) = 0$

$$\begin{aligned}
&\inf_{t\in[a,b]} \{|(Tu)(t)|, |(Tv)(t)|\} \\
&= \inf_{t\in[a,b]} \left\{ \left| \lambda \int_a^b K(t,s,u(s))\, ds \right|, \left| \lambda \int_a^b K(t,s,v(s))\, ds \right| \right\} \\
&\geq |\lambda| \inf_{t\in[a,b]} \left\{ \left| \int_a^b K(t,s,u(s))\, ds \right|, \left| \int_a^b K(t,s,v(s))\, ds \right| \right\} \\
&\geq |\lambda| \inf_{t\in[a,b]} \{|K(t,s,u(s))|, |K(t,s,v(s))|\} \int_a^b ds \\
&\geq |\lambda| L \inf_{t\in[a,b]} \{|u(t)|, |v(t)|\} \int_a^b ds \\
&= |\lambda| L(b-a) \inf_{t\in[a,b]} \{|u(t)|, |v(t)|\}
\end{aligned}$$

For λ such that $\alpha = |\lambda|L(b-a) \in (0,1)$, T is established to be a contraction.

Step 2: Apply the Similarity Fixed Point Theorem.

With T confirmed as a contraction on the complete similarity space $\mathcal{S}([a,b],\mathbb{R})$, the Similarity Fixed Point Theorem guarantees the existence of a unique fixed point $u^* \in \mathcal{S}([a,b],\mathbb{R})$ satisfying $Tu^* = u^*$, equating to a unique solution of the integral equation.

Thus, under the specified conditions, there exists a unique function $u \in \mathcal{S}([a,b],\mathbb{R})$ that solves the nonlinear Fredholm integral equation of the second kind. □

4.5.2 Application to Newton's Method

The method of Newton, or the Newton-Raphson method [168]–[170], stands as a paradigmatic example of numerical algorithms for root-finding, owing its robustness and efficacy to the foundational principles of fixed point theory. Specifically, Banach's Fixed Point Theorem provides a theoretical underpinning for the convergence properties of this method. This thesis presents a rigorous mathematical exploration of Newton's method applied to the task of root approximation utilizing similarity spaces and contraction mappings.

Consider a function $f : \mathbb{R} \to \mathbb{R}$, assumed to be continuously differentiable over its domain. The objective is to identify zeros of f, i.e., solutions to $f(x) = 0$, through iterative approximations. Newton's method facilitates this via the iterative scheme:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \tag{4.75}$$

where x_n denotes the n-th approximation to a root of f, and f' signifies the derivative of f.

To exemplify, the function is scrutinized $f(x) = x^2 - a$, with $a > 0$ and known to

possess at least two roots, notably $x = \pm\sqrt{a}$. Application of Newton's method as per Equation 4.75 yields the transformation:

$$Tx = x - \frac{x^2 - a}{2x} = \frac{1}{2}\left(x + \frac{a}{x}\right). \tag{4.76}$$

The function $T : [\sqrt{a}, \infty) \to [\sqrt{a}, \infty)$ is hereby established as a self-map under the condition that $x \geq \sqrt{a}$. Further, it holds that x is a fixed point of T if and only if $f(x) = 0$.

The convergence criterion leverages the mutual similarity condition, embodied in the inequality:

$$\begin{aligned} s(Tx, Ty) &= \min\{Tx, Ty\} \\ &\geq \frac{1}{2}\min\left\{x + \frac{a}{x}, y + \frac{a}{y}\right\} \\ &\geq \frac{1}{2}\min\{x, y\} + \frac{1}{2}\min\left\{\frac{a}{x}, \frac{a}{y}\right\} \\ &\geq \frac{1}{2}\min\{x, y\}, \end{aligned} \tag{4.77}$$

valid for all $x, y \in [\sqrt{a}, \infty)$. The self-similarity contraction condition is then

$$Tx \geq \frac{1}{2}x \tag{4.78}$$

Notably, the absence of additivity and multiplicativity in the minimum operation necessitates the adoption of inequality in the analysis. Consequently, T manifests as a contraction on the complete similarity space $([\sqrt{a}, \infty), |\cdot|)$, leading to the deduction, via the similarity contraction principle, that Equation 4.75 converges to the root $x = \sqrt{a}$ from any initiating guess $x_0 \in [\sqrt{a}, \infty]$.

The Newton's method is applied to find $a = \sqrt{2}$, starting from initial guesses $x_0 = 10$, $y_0 = 100$ and and $z_0 = 1000$. The following table displays the iteration steps for both initial guesses, showing the progression towards $\sqrt{2}$:

In numerical experiments, results are verified, and the current current speed of convergence of contraction mapping is calculated using $\alpha_1, \alpha_2, \alpha_3$:

- The mutual similarity speed of convergence is given by

$$\alpha_1 = \frac{\min\{Tx_n, Ty_n\}}{\min\{x_n, y_n\}}$$

- The self-similarity speed of convergence for x, y and z are expressed as

$$\alpha_2 = \frac{Tx_n}{x_n}, \alpha_3 = \frac{Ty_n}{y_n}, \alpha_4 = \frac{Tz_n}{z_n}.$$

Iteration	x_n	y_n	z_n	α_1	α_2	α_3	α_4
0	10	100	1000	-	-	-	-
1	5.1	50.01	500.001	0.5384	0.5384	0.5004	0.5
2	2.7461	25.0250	250.0025	0.6326	0.6326	0.5016	0.5
3	1.7372	12.5525	125.0052	0.8314	0.8314	0.5063	0.5001
4	1.4442	6.3559	62.5106	0.9794	0.9794	0.5248	0.5003
5	1.4145	3.3353	31.2713	0.9998	0.9998	0.5899	0.501

Table 4.1: Iteration steps for Newton's method with constants α_1, α_2, α_3, and α_4.

Concluding that all contractions α_1, α_2, α_3, and α_4 are greater than or equal to $\alpha = \frac{1}{2}$, as demnostrated in the Table 4.5.2.

Chapter 5

Entity Resolution in Similarity Space

5.1 Problem Formulation

Given a normalized similarity $s_n(\mathcal{R}_1, \mathcal{R}_2)$ is assumed, which measures the similarity between two records $\mathcal{R}_1$ and $\mathcal{R}_2$ as a real number in the range of $[0, 1]$, with 1 indicating complete similarity and 0 indicating complete dissimilarity. A fixed α, where $0 < \alpha < 1$, is considered. The distinction of relevant records into two classes by the applications, namely the set of matches ($\mathcal{M}$) and the set of non-matches ($\mathcal{N}$), consisting of ordered pairs of records $(\mathcal{R}_1, \mathcal{R}_2)$, is achieved through a binary classifier. More formally,

$$\begin{aligned} (\mathcal{R}_1, \mathcal{R}_2) &\in \mathcal{M} \quad \text{if } s_n(\mathcal{R}_1, \mathcal{R}_2) \geq \alpha, \\ (\mathcal{R}_1, \mathcal{R}_2) &\in \mathcal{N} \quad \text{if } s_n(\mathcal{R}_1, \mathcal{R}_2) < \alpha. \end{aligned} \tag{5.1}$$

The decision where to set the match/non-match threshold α is a balancing act. The choice should be based on an acceptable sensitivity (or recall, the proportion of truly matching records that are linked by the algorithm) and positive predictive value (or precision, the proportion of records linked by the algorithm that truly do match). The task is to find an algorithm that classifies the sets of matches and non-matches as accurately as possible corresponding to reality as judged by human observation. In the scientific literature, the terms *record linkage* [171]–[173], *entity linking* [174], *data matching* [130], *fuzzy matching* [175], *entity resolution* [176], [177], etc., are also encountered.

In information retrieval theory and search engines, the phrase 'retrieving relevant documents' can analogically be met with. A record can be transformed into tokens of words. That is, each record is split into a set of tokens $\{X_1, \ldots, X_j, \ldots, X_n\} \in \mathcal{R}_1$. The size of a record $\mathcal{R}_1$ is denoted by $|\mathcal{R}_1|$, which represents the number of tokens in $\mathcal{R}_1$ [171]

5.2 State-of-the-Art

The goal is to determine the similarity of the two strings. The similarity can be semantic or syntactic. Strings are semantically similar if they have the same meaning, for example, car and automobile, and syntactically similar if they have the same sequence of characters [178].

From biological perspective the information of the people is stored imprecisely in the brain - in the brain noise. The imprecise information based on modeling human traits similarity should be incorporated in current search engines. The approximate string matching algorithms have been developed to solved imprecise similarity between records due misspelling, typographical, phonetic error, wrong input, misinterpretation and others.

For a short overview, the approximate string matching algorithms could be grouped by similar features into categories such as character-based similarity (phonetics, heuristics, Q-gram, edit-based) and token-based similarity [178]–[180]. In this thesis, a novel class of character-based category, convolution-based, is introduced, as seen in Figure Figure 5.1.

5.2.1 Character-Based Similarity

Phonetic Similarity

Into group of phonetics algorithms belong Soundex [181], [182], Metaphone [183], Double Methaphone [184], Phonex [185], Phonix [186], NYSIIS [187], or Fuzzy Soundex [188]. These algorithms are characterized by a strong linguistic phonetic dependence (usually English) which leads to poor F-measure results [189], poor robustness against misspelling and typographical errors and they don't have assumptions of tokens. Due to language specificity they cannot be well used on multiple-language data sources. However they are still implemented in many databases and software applications of leading software companies. Misunderstanding of the usage of such algorithms for various languages for which they are not intended is very often seen.

Heuristic Similarity

Heuristics are those that do not have a solid theoretical basis and have been derived empirically. Jaro distance [190] and further its modification Jaro-Winkler [191] distance are well-known representatives and very often used in the commercial sector for their simple implementation and high F-measure [130], [180], [189]. They are appropriate only for short string, e.g. names and don't have an assumption of tokens, sensitive for disorder of a sequence of tokens.

Edit-Based Similarity

The edit distance between two strings is the minimum cost of edit operations need to transform one string into another. Each operation has a cost function associated often with the cost 1. Here are well-know algorithms based on dynamic programming, Levenshtein [123], Damerau-Levenshtein [192], [193], for DNA or protein sequence comparison following Needleman-Wunsch [69], Smith-Waterman [70], Gotoh [194] etc. Here could be mentioned also learning cost distance with supervised machine learning for specific transitions [195]. Unfortunately, these standard similarity are not practically usable for bag of words model, where also permutation of words should be involved. The other issue is also normalization, e.g. Levenshtein distance in a measurement absolute edit distance between terms, is not practically usable for their bias for words with different length. The normalization factors are proposed [111], [196], [197] that normalizes Levenshtein distance in range $[0; 1]$. However, some of factors violates similarity axioms, especially triangle inequality. In structured documents is implicit information of the context given by columns as a semantic topic. Each attribute is modeled using bag of words model, where the context very often is not relevant in traditional databases and NoSQL storages.

Q-grams

The Q-grams (also called 'n-grams') are short character substrings of length q of the record strings [198]. They have been applied, in many variations, e.g. different spelling correction methods. or inverted indexes [177], [199], [200]. Q-gram method belongs to the most favourite indexing method in enterprise and web search engines used by tech companies with the largest market share and capitalization. The comparison with other similarities has quite good results, however token-based algorithms outperforms this method [189]. It can be explained that Q-grams do not treat natural language encoding based on words (with "tokens" being used in the text for greater generality), but rather, they split tokens into "artificial" substrings, ignoring the information regarding from which token the Q-gram was generated. This concept achieves good results for robustness of disorder token sequence but at overall accuracy performance are used rather as indexing or filtering methods. They serve so as a preprocessing step for more accurate approximate string matching algorithms.

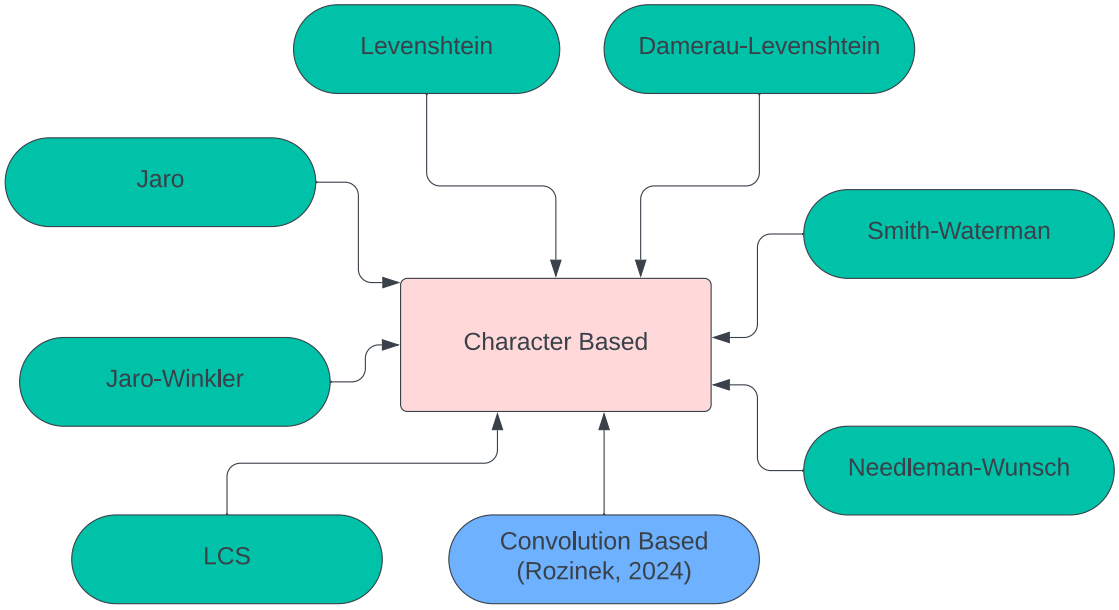

Figure 5.1: This figure delineates the new taxonomy of character-based approximate string matching algorithms, categorizing them by their methodological approach and operational characteristics. A newly introduced class, termed 'Convolution-Based', as discussed in this thesis, highlights the main contribution.

5.2.2 Token-Based Similarity

In structured documents, there is implicit information from the context, given by columns as a semantic topic. Each attribute is modeled using a bag-of-words model, where the context very often is not relevant in traditional databases and NoSQL storage. The previously mentioned methods are based on the comparison of characters in the complete string. A hybrid similarity method converts the string into a set of tokens, the so-called bag-of-words model. A string can be transformed into sets by splitting using a delimiter. This allows taking into account the semantic meaning of the words and processing large texts. The comparison has two levels: a character level comparison for each pair of tokens, and a comparison on the token level of all possible pairs of tokens. The similarity calculation is usually based on the Jaccard index, or the Sorensen–Dice formulae. For the token set there are used comparisons such as Monge-Elkan [201], [202] or SofTFIDF [130]. Unfortunately, these methods perform an approximate combinatorial assignment, and hence yield an asymmetric matching, see the example below. Maximum matching on bipartite graph [203]–[205] is more suitable for practical use and achieves the best F-measures due to its solving the optimal assignment problem and the symmetricity of their measurements. The great advantage of these methods is that they respect the encoding of natural language with respect to token resolution as words. It can be found that this

group of models deserves the most attention in research for its promising results.

Unfortunately, those algorithms often have a quadratic time complexity $\mathcal{O}(n^2)$ or even a cubic time complexity $\mathcal{O}(n^3)$ or worse $\mathcal{O}(n^4)$, especially those providing the best recall/precision results on research data sets.

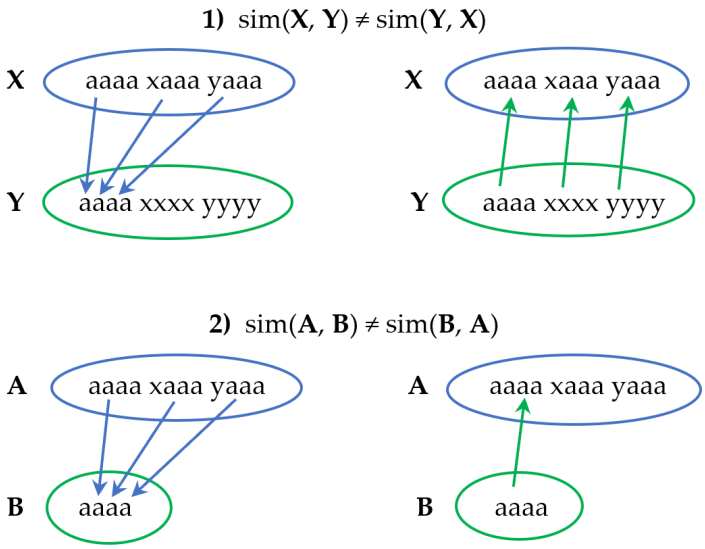

Figure 5.2: Two examples of an asymmetric Monge–Elkan measure sim_{ME}.

Example of Asymmetric Matching

Monge and Elkan [201] proposed recursive field matching, which measures the similarity distance between two records $\mathcal{R}_1$ and $\mathcal{R}_2$. Each string is broken up into token sets $\mathcal{R}_1 = \{X_1, X_2, \ldots, X_n\}$ and $\mathcal{R}_2 = \{Y_1, Y_2, \ldots, Y_m\}$. Then the similarity is expressed by

$$\mathrm{sim}_{ME}(\mathcal{R}_1, \mathcal{R}_2) = \frac{1}{n} \sum_{i=1}^{n} \max_{j=1}^{m} sim'(X_i, Y_j), \tag{5.2}$$

where sim′ is an internal similarity function able to calculate the similarity between two single tokens. This is a measure that is independent of the sequential order of the tokens. This equation approximates the solution of the combinatorial assignment problem in combinatorial optimization. Unfortunately, the Monge–Elkan approximation is not a symmetrical similarity function, see Figure 5.2.

In this survey, the most cited algorithms based on fuzzy set relatedness have been identified by me [206]. Hence, they are referred to as the current state of the art. Readers are referred to the commonly used similarity functions, namely fuzzy-dice similarity, fuzzy-cosine similarity, and fuzzy Jaccard similarity, by me [204]–[206]. The proposed approximate record matching method will be based on this class of algorithms.

Definition 24 (Fuzzy-Token Similarity [204], [205]). *At the disposal are two sets of tokens,* $\mathcal{R}_1$ *and* $\mathcal{R}_2$*. Write* $\mathcal{R}_1 \widetilde{\cap}_\delta \mathcal{R}_2$ *for the fuzzy overlap of* $\mathcal{R}_1$ *and* $\mathcal{R}_2$*. The value* δ *is the token level threshold in the internal similarity function* $\mathrm{sim}'(X_i, Y_j)$*. The incident edge for the token pair is considered only for* $\mathrm{sim}'(X_i, Y_j) \geq \delta$*. The following similarity functions will be defined:*

Fuzzy Dice Similarity, $\mathrm{sim}_D(\mathcal{R}_1, \mathcal{R}_2)$*,*

$$\mathrm{sim}_D(\mathcal{R}_1, \mathcal{R}_2) = \frac{2|\mathcal{R}_1 \widetilde{\cap}_\delta \mathcal{R}_2|}{|\mathcal{R}_1| + |\mathcal{R}_2|}. \tag{5.3}$$

Fuzzy Cosine Similarity, $\mathrm{sim}_C(\mathcal{R}_1, \mathcal{R}_2)$*,*

$$\mathrm{sim}_C(\mathcal{R}_1, \mathcal{R}_2) = \frac{|\mathcal{R}_1 \widetilde{\cap}_\delta \mathcal{R}_2|}{\sqrt{|\mathcal{R}_1|}\sqrt{|\mathcal{R}_2|}}. \tag{5.4}$$

Fuzzy Jaccard Similarity, $\mathrm{sim}_J(\mathcal{R}_1, \mathcal{R}_2)$*,*

$$\mathrm{sim}_J(\mathcal{R}_1, \mathcal{R}_2) = \frac{|\mathcal{R}_1 \widetilde{\cap}_\delta \mathcal{R}_2|}{|\mathcal{R}_2| + |\mathcal{R}_1| - |\mathcal{R}_1 \widetilde{\cap}_\delta \mathcal{R}_2|}. \tag{5.5}$$

Fuzzy Overlap Similarity, $\mathrm{sim}_J(\mathcal{R}_1, \mathcal{R}_2)$*,*

$$\mathrm{sim}_O(\mathcal{R}_1, \mathcal{R}_2) = \frac{|\mathcal{R}_1 \widetilde{\cap}_\delta \mathcal{R}_2|}{\min\{|\mathcal{R}_1|, |\mathcal{R}_2|\}}. \tag{5.6}$$

The relaxed term 'similarity function' is used instead of 'normalized similarity' because, e.g. Dice similarity is not a normalized similarity, as can be simply deduced from [207].

In the definition of fuzzy token similarity, a fuzzy overlap is added compared to the original articles, as the author will continue to be worked with. However the previously introduced models have some disadvantages from the perspective:

- *Second Threshold* The disadvantage is that two threshold parameters are actually present – the threshold δ in the internal similarity function $\mathrm{sim}'(X_i, Y_j)$ and the overall threshold α on $\mathrm{sim}(\mathcal{R}_1, \mathcal{R}_2)$ with these, the classification of whether the

records $\mathcal{R}_1$ and $\mathcal{R}_2$ are a match or not is determined. The global threshold α can be quite different from the local threshold δ. In addition, the threshold of the internal function δ requires optimization and can lead to reduced accuracy if selected incorrectly.

- *Not Entirely Fuzzy* By applying a binary classifier $\text{sim}'(X_i, Y_j) \geq \delta$, whether a token pair (X_i, Y_j) is classified as a match. Such a strict classification on the token level leads to declining matches on tokens lower than the threshold δ and does not respect human-natural continuous perception of the overall similarity, because the information about any similarity lower than δ is replaced by a substitution value of zero. Further, see an example in [208] – *Fig. 2. Bipartite graphs with different* δ, where some edges in the weighted bipartite matching are deleted due to the thresholding. A sub-optimal solution may be obtained due to the lack of a complete bipartite graph. With this decision, a Bayes risk in Bayesian decision theory is also accepted [90], [209].

- *Not a similarity* The triangle inequality given by (N2), respectively (S2), is violated. This statement is supported by using the normalized Levenshtein similarity, which violates the triangle inequality, as further mentioned in the next subchapter.

  Example It is known that a normalized similarity is given by the relation $s_n(x, y) = 1 - d_n(x, y)$. Hence, it is only proven that the term $d_n(x, y) = \frac{d(x,y)}{\max\{|x|,|y|\}}$ does not qualify as a distance metric. The edit distance is a solution of a dynamic programming problem and can only be obtained numerically. The task of the embeddability of the edit distance into l_p norms is still an open problem. It has been shown that such a metric cannot be embedded into the l_1 norm (with arbitrary dimension) with distortion better than $\frac{3}{2}$ [210].

  Proof. Let's illustrate this rather with the example of the strings $X =$ "*ab*", $Y =$ "*abc*" and $Z =$ "*bc*". Then, it is obtained that

$$\begin{aligned} &d(X, Z) \leq d(X, Y) + d(Y, Z), \\ &\frac{d(X, Z)}{max\{|X|, |Z|\}} \leq \frac{d(X, Y)}{max\{|X|, |Y|\}} + \frac{d(Y, Z)}{max\{|Y|, |Z|\}}, \\ &\frac{2}{2} \not\leq \frac{1}{3} + \frac{1}{3}. \end{aligned} \tag{5.7}$$

  Assertion, it is shown that this normalization does not preserve the properties of metric space. □

5.2.3 Deep Learning for Matching

One approach to dealing with string matching is shown by [211], who present their software library for fuzzy string matching and candidate ranking; their approach supports different deep neural network architectures to train new classifiers and to tune the trained data. This thesis presents their approach to building the library and tuning the training dataset. They also compare the results with existing systems such as [212], which however relies on lookup tables. They compare their results with [213], who bring an approach called toponym matching, which is used, for example, in matching paired texts that represent the same real locations and are therefore often used in the geosciences. [213] use the approach of Gated Recurrent Units [214], a recurrent neural network used in modeling sequence data. An example is a representation of a sequence of bytes corresponding to texts with which they have something in common. Their approach uses the deep neural network architecture proposed by [134], where the parameters are learned from the training data. The whole model can then be trained end-to-end using a back-propagation algorithm following an optimization method proposed by [215] and called Adam. The thesis then shows results on a large dataset collected by the GeoNames gazetter from www.geonames.org.

Another approach is discussed by [216], which shows ontology matching based on deep neural networks, noting that there are approaches that ignore higher-level correlations between different descriptions of the entities under test. Therefore, they propose learning methods that take these approaches into account. [216] argue that ontology matching usually measures the similarity between two entities from two different ontologies, where a pair of entities with high similarity is called a mapping. To correctly exchange data between ontology-based applications, it is important to establish correspondences (or mappings) between their ontologies. However, creating such mappings manually is very difficult due to the complexity of modern ontologies. A deep learning model is then a tool that can be used to learn a high-level abstract representation of the original data. Their study used ontology matching to solve the semantic heterogeneity problem and focused on searching different ontologies among semantically similar entities.

Convolutional Approaches

Character-level convolutional neural networks (CNNs): These have been explored for string similarity, achieving high accuracy but often requiring large training datasets and computational resources [88], [131], [217].

Word embeddings with cosine similarity: Embedding methods learn vector representations of words, and cosine similarity measures the angle between them [218]. While

effective for semantic similarity, they might not capture precise character-level differences relevant for this task [178].

Limitations of Existing Approaches: Edit distance measures are generally computationally expensive for large datasets [219]. Character-based metrics like Jaro and Jaro-Winkler neglect the importance of character order within the matching window. While CNNs offer high accuracy, they can be resource-intensive [88], [218]. Word embeddings might not be optimal for capturing fine-grained character-level similarities [178].

5.3 Convolution-Based String Matching

Character-based similarity metrics represent critical tools in various domains, including data mining, bioinformatics, and natural language processing. Widely used methods such as Jaro [190] and Jaro-Winkler [191], [220]–[222] offer efficient similarity evaluation, but they often neglect the significance of character order within the matching window. This disregard can potentially compromise accuracy, especially when dealing with strings where specific sequence holds considerable importance.

In this thesis, two novel similarity measurements are proposed, Convolutional Jaro (ConvJ) and Convolutional Jaro-Winkler (ConvJW) [OR-5], that effectively address this limitation. Both similarity measures leverage a convolutional approach with a Gaussian weighting function to effectively capture the positional proximity of matching characters, leading to significantly enhanced accuracy compared to existing methods in the category of unsupervised character based approximate string matching.

The key contributions of this work comprise:
Improved accuracy: ConvJ and ConvJW achieve superior accuracy as measured by F1-score, surpassing the state-of-the-art in unsupervised character based approximate string matching.
Computational efficiency: Both metrics maintain computational efficiency comparable to Jaro and Jaro-Winkler, making them suitable for practical applications.
Faster execution: ConvJ exhibits even faster execution times compared to the standard Jaro implementation.

These characteristics render ConvJ and ConvJW ideal for tasks requiring high-performance string similarity calculations. They set the stage for continued investigation of convolutional similarity measures across diverse fields.

Despite different modern techniques the leading technological companies using Jaro and Jaro-Winkler in their products and it belongs to the most impactful and most used similarity function for task in deduplication and matching. Some large products where

such techniques are implemented are listed:

Commercial Products

- *Oracle Data Quality*: A component of Oracle's broader data management suite that offers data cleansing, profiling, and matching capabilities. It likely uses algorithms similar to Jaro-Winkler for its matching and deduplication processes to ensure high data quality across enterprise systems [223].
- *Talend Open Studio*: This open-source data integration platform includes the Jaro-Winkler similarity function in its Data Profiling and Data Quality, which can be used for data cleansing and matching. You can find the documentation for the Data Profiling and Data Quality solution here [224].
- *IBM InfoSphere* This data integration platform includes the Jaro-Winkler similarity function in its Transformer stage, which can be used for data cleansing and matching. You can find the documentation for the Transformer stage here [225].
- *Informatica Data Quality* This data integration platform includes the Jaro and Jaro-Winkler similarity functions in its Data Quality transformation, which can be used for data cleansing and matching. You can find the documentation for the Data Quality transformation here: [226].

Open-source Libraries

- *Apache Lucene* This open-source search engine library includes the Jaro-Winkler similarity function in its FuzzyQuery class, which can be used for fuzzy search queries. You can find the documentation for the FuzzyQuery class here: [227].

This list is far from exhaustive; there are numerous other commercial products and open-source libraries that utilize the Jaro or Jaro-Winkler algorithms.

Addressing the Limitations: ConvJ addresses these limitations by combining the efficient matching mechanism of Jaro with a convolutional approach. It incorporates Gaussian weighting to emphasize matches closer in position and leverages convolutions to capture the overall similarity landscape effectively. This results in a computationally efficient metric that considers both character matches and their relative order, leading to improved accuracy compared to existing methods.

5.3.1 Convolution-Based String Similarity Model

This section introduces a generalized model for convolution-based string similarity measures, extending beyond specific instances like Jaro to a broader class of similarity functions. By employing a generic matching kernel, this model provides a flexible foundation for developing and analyzing various string matching algorithms through convolution principles.

In the context of string similarity, convolution can be conceptualized as the process of sliding a matching kernel over one string (considered as the signal) to evaluate its similarity against another string (treated as the filter). Mathematically, this process is represented by the convolution sum:

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n-m], \tag{5.8}$$

where $f[n]$ corresponds to the character sequence of the first string S_1, and $g[n]$ represents the generic matching kernel applied to S_1 and the second string S_2. This formulation captures the essence of convolution-based string similarity by systematically evaluating the match between characters across the two strings.

A matching kernel, $g[n]$, is a function that quantifies the similarity between characters from two strings within a defined window of comparison. The choice of kernel is pivotal, as it determines the sensitivity of the similarity measure to various factors, such as positional alignment, character equivalence, and the proximity of matches. For a broad class of convolution-based similarity functions, kernels can vary significantly, including, but not limited to:

- Gaussian kernels for weighted positional similarity,
- Kronecker delta functions for exact character match emphasis,
- Custom kernels designed to capture specific linguistic or domain-specific properties.

The flexibility in kernel selection allows for tailoring the similarity measure to the specific requirements of the application domain, enhancing both the accuracy and relevance of the comparisons.

Definition 25 (Generalized Convolution-Based String Similarity)**.** *Let Σ be a finite alphabet, and let Σ^* denote the set of all finite strings over Σ. The generalized convolution-based string similarity measure $s_{conv}(S_1, S_2) \colon \Sigma^* \times \Sigma^* \to \mathbb{R}_{\geq 0}$ quantifies the similarity between S_1 and S_2 utilizing a convolutional approach. Formally, $s_{conv}(S_1, S_2)$ is defined by the equation:*

$$s_{conv}(S_1, S_2) = \sum_{i=0}^{|S_1|-1} \sum_{j=0}^{|S_2|-1} G(i,j) \cdot K(S_1[i], S_2[j]), \tag{5.9}$$

where:

- $G(i,j)$ *is a weighting function that modulates the importance of matches based on their positional characteristics, thereby facilitating the incorporation of spatial information into the similarity calculation.*
- $K(S_1[i], S_2[j])$ *is a kernel function that evaluates the similarity between the* i*-th character of* S_1 *and the* j*-th character of* S_2*. The choice of kernel function* K *allows for the customization of the similarity measure to accommodate various matching criteria and character equivalences.*

This framework establishes a new class of convolution-based string similarity measurements, grounded in a mathematical model that extends the utility of convolution operations to approximate string matching. Central to this model is the flexibility in kernel function selection, enabling the development of similarity measurements tailored to specific data characteristics and analysis requirements. This methodological advancement supports the creation of similarity measures that are both precise and adaptable, suitable for a diverse range of applications from text processing to bioinformatics.

By incorporating various kernel functions, the model provides a robust foundation for capturing complex patterns of similarity, including character substitutions and phonetic variances. This allows for a deeper analysis of character-based data, enhancing the accuracy and efficiency of similarity computations. The framework's adaptability to different kernel functions opens up possibilities for exploring novel approaches to approximate string matching.

5.3.2 Convolutional Jaro (ConvJ)

The Jaro similarity measure [190], introduced by Matthew A. Jaro, serves as a measure for evaluating the similarity between two strings, denoted as $S1$ and $S2$. This metric is particularly useful in fields such as record linkage and spell checking, quantifying the degree of similarity on a scale from 0 to 1, where 0 indicates no similarity and 1 denotes an exact match.

Definition 26 (Jaro Similarity [190], [191])**.** *Let* Σ *be a finite alphabet, and let* Σ^* *denote the set of all finite strings over* Σ*. The Jaro similarity measure* $s_J \colon \Sigma^* \times \Sigma^* \to [0,1] \subset \mathbb{R}$ *between two strings* S_1 *and* S_2 *is defined as follows. Let* d *be the maximum distance*

within which characters from one string can match with characters from the other string, determined by:

$$d = \left\lfloor \frac{\max(|S_1|, |S_2|)}{2} \right\rfloor - 1. \tag{5.10}$$

The number of matching characters m between S_1 and S_2 is defined as the maximum number of one-to-one correspondences between characters in S_1 and S_2 that can be made such that for any matched pair $(S_1[i], S_2[j])$, $|i - j| \leq d$. This captures the essence of forming matches within a sliding window in the sequence, ensuring each character from S_1 is matched uniquely to a character in S_2 within the allowable distance and vice versa.

The number of transpositions t is calculated based on the sequence of matching characters, counting instances where matched characters are in a different order in the two strings.

The Jaro similarity measure is then calculated by:

$$s_J(S_1, S_2) = \begin{cases} 0 & \text{if } m = 0, \\ \frac{1}{3}\left(\frac{m}{|S_1|} + \frac{m}{|S_2|} + \frac{m - t/2}{m}\right) & \text{otherwise.} \end{cases} \tag{5.11}$$

The computational complexity of the Jaro similarity Algorithm 2 is $\mathcal{O}(|S1||S2|)$, where $|S1|$ and $|S2|$ are the lengths of the two input strings. This efficiency makes it highly suitable for real-time data processing and large-scale data matching tasks. The Jaro similarity measure has found extensive applications in various domains requiring accurate string comparison, including database cleaning, information retrieval, and natural language processing. It also serves as the foundation for more sophisticated similarity metrics, such as the Jaro-Winkler distance, which introduces adjustments for common prefixes to increase precision in specific contexts.

The ConvJ is innovative enhancements of the traditional Jaro similarity algorithm, incorporating a convolutional approach with Gaussian weighting to assess the similarity between two strings. This method extends the original Jaro algorithm by applying a Gaussian-weighted convolution operation, aiming to capture the positional proximity of matching characters with greater detailed similarity evaluation. See Algorithm 4.

For any two positions i and j within strings S_1 and S_2, respectively, the Gaussian weight is computed as:

$$G(i, j) = \exp\left(-\frac{|i - j|^2}{2\sigma^2}\right) = \exp\left(-\frac{d^2}{2\sigma^2}\right), \tag{5.12}$$

where σ represents the standard deviation of the Gaussian kernel. This weighting function decreases the influence of character matches as their positional distance increases, with σ controlling the rate of this decrease.

The convolution operation aims to identify the optimal character match between strings S_1 and S_2 within a predefined sliding window, utilizing Gaussian weighting to balance proximity and character accuracy. This procedure is encapsulated by the calculation of M_w, representing the accumulated sum of optimal match evaluations across all character positions in S_1:

$$M_w = \sum_{i=0}^{|S_1|-1} \max_{j \in J(i)} \left\{ G(i,j) \cdot \delta_{S_1[i],S_2[j]} \right\}, \tag{5.13}$$

where $G(i,j)$ denotes the Gaussian weight function, emphasizing the impact of positional differences between characters at indices i in S_1 and j in S_2. The Kronecker delta function, $\delta_{S_1[i],S_2[j]}$, indicating an exact match between characters at positions i and j:

$$\delta_{S_1[i],S_2[j]} = \begin{cases} 1 & \text{if } S_1[i] = S_2[j], \\ 0 & \text{otherwise.} \end{cases} \tag{5.14}$$

The index set $J(i)$ defines the matching range for a character at position i, determined by:

$$J(i) = \{ j : \max(0, i-w) \leq j \leq \min(|S_2|-1, i+w) \}, \tag{5.15}$$

with w representing the half-window size, calculated based on the desired coverage percentage and the standard deviation (σ) of the Gaussian distribution. The calculation of w is informed by the z-score (Z), which is derived from the desired coverage percentage in a normal distribution. For a coverage of 99.9%, the z-score is determined as follows

$$Z = \Phi^{-1}\left(\frac{99.9\% + 1}{2}\right), \tag{5.16}$$

where Φ^{-1} denotes the inverse of the cumulative distribution function (CDF) for a standard normal distribution. Given $Z \approx 3.29$ for 99.9% coverage, w can be calculated by:

$$w = \lceil 3.29 \cdot \sigma \rceil, \tag{5.17}$$

where $\lceil \cdot \rceil$ denotes the ceiling function, ensuring that w is sufficiently large to include at least 99.9% of the normal distribution's weight, thus maximizing the likelihood of capturing the most relevant character matches within the window. This selection of w ensures that the convolution operation robustly accounts for both the precision of character matches and their positional proximity.

The ConvJ similarity measure enhances string similarity evaluation by incorporating a

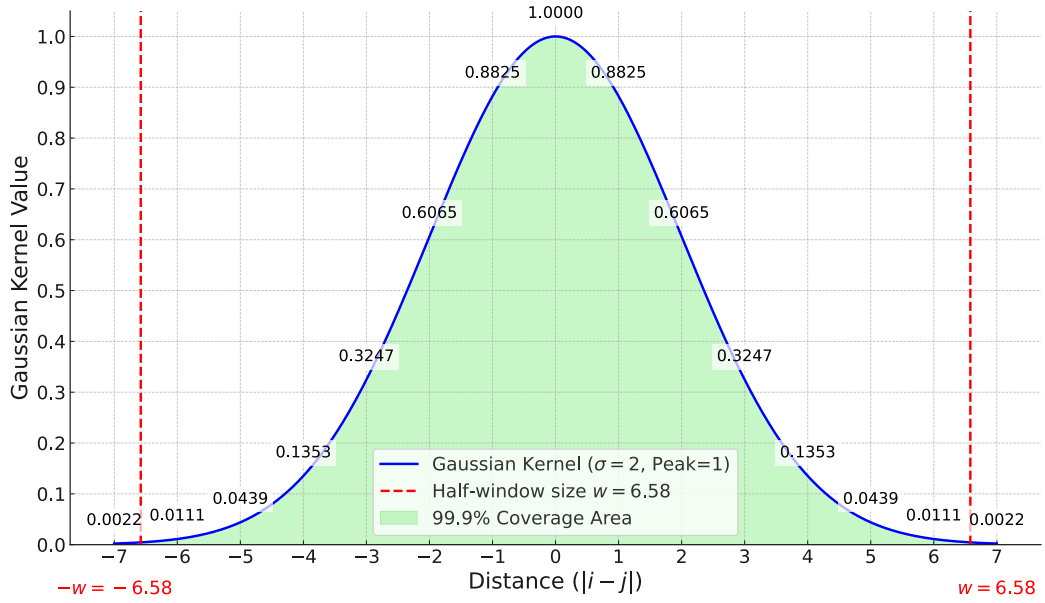

Figure 5.3: The graph illustrates the Gaussian kernel's impact on convolution-based string similarity, with a standard deviation ($\sigma = 2$), highlighting the 99.9% coverage area critical for the ConvJ and ConvJW algorithms. The graph, marked with $-w$ and w to denote the half-window size, emphasizes the balance between character match precision and proximity.

detailed assessment of character misalignment between strings S_1 and S_2. This approach extends beyond simple transposition counts to offer a granular examination of character positional deviations and their impact on similarity perception.

Misalignment is assessed using an adjusted inverse Kronecker delta function, $\bar{\delta}_{ij}$, to penalize positional discrepancies more effectively:

$$\bar{\delta}_{ij} = 1 - \delta_{ij}, \tag{5.18}$$

where δ_{ij} is defined as:

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases} \tag{5.19}$$

indicating exact alignment of characters at index i in S_1 with index j in S_2, and 0 for misalignments. The complement, $\bar{\delta}_{ij}$, quantifies the degree of misalignment, which is then weighted by the Gaussian function, $G(i, j)$, to account for the significance of positional differences:

$$A_w = \sum_{(i,j)\in M} \bar{\delta}_{ij} \cdot G(i, j), \tag{5.20}$$

Here, A_w denotes the cumulative weighted sum of misalignments across all character

position pairs within the set M. This measure not only identifies character matches but also captures the complex spatial dynamics of string similarity, enhancing the precision of the similarity score. The set M is defined as the collection of all character position pairs (i, j) where character i from string S_1 is evaluated against character j from string S_2. Formally, M can be represented as:

$$M = \{(i, j) : i \in S_1, j \in S_2, \text{ and } |i - j| \leq w\}. \tag{5.21}$$

The parameter σ in the Gaussian weighting function adjusts the sensitivity to positional differences. A larger σ broadens the Gaussian distribution, accommodating misalignments over larger distances and thus penalizing distant mismatches less severely. In contrast, a smaller σ emphasizes immediate proximity, focusing the assessment on closely aligned character pairs. This flexibility allows for the algorithm to be tailored to different application needs, balancing precision with positional variance tolerance.

This approach ensures that the ConvJ similarity measure not only quantify character matches but also account for both character presence and their orderly sequence.

Summarizing these derivations leads to the final definition of ConvJ.

Definition 27 (ConvJ [OR-5])**.** *Given a finite alphabet Σ and Σ^* denoting the set of all finite strings over Σ, the ConvJ similarity measure $s_J : \Sigma^* \times \Sigma^* \to [0, 1] \subset \mathbb{R}$ between two strings S_1 and S_2 utilizes a convolutional approach with Gaussian weighting to evaluate character matches with attention to their positional proximity. For any two positions i and j within strings S_1 and S_2, respectively, a Gaussian weight is computed as:*

$$G(i, j) = \exp\left(-\frac{(i - j)^2}{2\sigma^2}\right), \tag{5.22}$$

where σ is the standard deviation of the Gaussian kernel. The ConvJ similarity incorporates a convolution operation to identify optimal character matches within a predefined sliding window, weighted by $G(i, j)$, to compute the similarity score as follows:

$$s_{ConvJ} = \frac{1}{3}\left(\frac{M_w}{|S_1|} + \frac{M_w}{|S_2|} + \frac{M_w - A_w}{M_w}\right), \tag{5.23}$$

where M_w represents the sum of Gaussian-weighted matches, and A_w denotes the weighted sum of misalignments between S_1 and S_2. The procedure calculates M_w by accumulating the maximum of $\{G(i, j) \cdot \delta_{S_1[i],S_2[j]}\}$ for each character position in S_1, and A_w accounts for positional discrepancies between matching characters.

The example Equation B.1 illustrates how ConvJ assesses string similarity, taking into account both the convolutional matching strategy and the significance of starting characters, especially when applying Gaussian weighting with specific σ and w parameters.

As a reference implementation of the Jaro similarity for the experiments, the Rosetta Code implementation [228] (see Algorithm 2) in the C# programming language is utilized. Several enhancements are made to the implementation to improve execution time.

The proposed ConvJ, detailed in Algorithm 4, introduces several optimizations over the standard Jaro similarity calculation, leading to significant improvements in computational efficiency, accuracy and execution speed. Key enhancements include:

1. **Elimination of Match Tracking Arrays:** Unlike the original Jaro algorithm, which utilizes boolean arrays to track character matches between strings (lines 7-8 of Algorithm 2), *ConvJ* foregoes this approach. This optimization reduces memory overhead and eliminates the need for multiple array access operations, thereby enhancing runtime performance.

2. **Integrated Match and Miasalignment Calculation:** ConvJ computes character matches and misalignments simultaneously within a single iteration through each string. This integration contrasts with the Jaro method's sequential process, which separately identifies matches and then calculates transpositions, thus reducing the overall computational steps required. Through the loop in lines 24–27 of Algorithm 4, matches for each character in S_1 are sought in S_2 within the window w, thereby optimizing both match detection and misalignment evaluation.

3. **Precomputed Gaussian Weights:** The algorithm leverages a precomputed vector of 1D Gaussian weights based on the relative character positions within a predefined window size. This precalculation avoids repetitive weight computations during runtime, leading to a more efficient execution. These values are precomputed based on the window size w and standard deviation σ (lines 1–3 and the *PrecomputeGaussian* function in lines 4–9 of Algorithm 4).

4. **Localized Comparison with Gaussian Weighting:** By applying Gaussian weights to character comparisons, ConvJ emphasizes closer character matches over distant ones. This approach not only aligns with the intuitive understanding of string similarity but also minimizes unnecessary computations for characters outside the maximum window size, further speeding up the algorithm.

5. **Early Termination on Perfect Character Match:** If a perfect match (character equality with maximum Gaussian weight) is discovered, the algorithm terminates the inner loop early (line 20 of Algorithm 4), sidestepping unnecessary comparisons. This optimization proves particularly efficacious for strings with high similarity, curtailing the average computation time.

These improvements collectively contribute to a more performant execution of the ConvJ similarity measurement, making it particularly suitable for applications requiring high-throughput processing of string comparisons.

5.3.3 Convolutional Jaro-Winkler (ConvJW)

The Jaro-Winkler similarity measure [191] enhances the Jaro similarity measure by giving more favorable scores to strings that match from the beginning for a set prefix length. This adjustment is particularly beneficial in applications where common prefixes are an indication of similarity, such as name matching in record linkage. The Algorithm 3 was developed by William E. Winkler [191] to improve the accuracy of the Jaro similarity metric in certain contexts.

The Jaro-Winkler similarity score, $s_{\mathrm{JW}}(S1, S2)$, is calculated using the Jaro similarity score, $s_J(S1, S2)$, with an additional boost for common prefixes. The formula is given by:

$$s_{JW}(S1, S2) = s_J(S1, S2) + l \cdot p \cdot (1 - s_J(S1, S2)), \tag{5.24}$$

where l is the length of the common prefix up to a maximum of 4 characters, p is a scaling factor for how much the score is adjusted upwards for prefix similarity. A typical value for p is 0.1.

The common prefix length l is defined as the number of characters from the start of the strings that are identical, up to a maximum of 4 characters. This means that the maximum possible adjustment to the Jaro similarity score is $0.1 \times 4 \times (1 - s_J(S1, S2)) = 0.4 \times (1 - s_J(S1, S2))$, which can significantly influence the final similarity score for strings with common prefixes. See Equation B.1.

The Jaro-Winkler similarity [191] algorithm maintains the computational efficiency of the Jaro similarity, with an additional step to calculate the prefix length. This makes it equally suitable for real-time applications and large datasets where precise string matching is crucial. The introduction of the prefix scaling factor enhances the matching accuracy for strings with common beginnings, making it especially useful in the fields of data deduplication, record linkage, and information retrieval where such characteristics are common.

Definition 28 (ConvJW [OR-5])**.** *Given a finite alphabet Σ and Σ^* denoting the set of all finite strings over Σ, the ConvJW similarity measure $s_{ConvJW} : \Sigma^* \times \Sigma^* \to [0, 1] \subset \mathbb{R}$ between two strings S_1 and S_2 builds upon the ConvJ similarity by integrating exponential decay to weight the significance of matching characters based on their position in the strings, with the intent to emphasize matches closer to the beginning of the strings. For any two positions i and j within strings S_1 and S_2, respectively, a Gaussian weight is*

computed as:

$$G(i,j) = \exp\left(-\frac{(i-j)^2}{2\sigma^2}\right), \tag{5.25}$$

where σ is the standard deviation of the Gaussian kernel, and an exponential decay factor is applied as:

$$E(i,j) = \exp(-\beta \cdot \min(i,j)), \tag{5.26}$$

with β representing the decay rate. The ConvJW similarity incorporates these weights in a convolution operation to compute the similarity score as follows:

$$s_{ConvJW} = \frac{1}{3}\left(\frac{M_w}{C[|S_1|]} + \frac{M_w}{C[|S_2|]} + \frac{M_w - A_w}{M_w}\right),$$

where M_w represents the sum of Gaussian-weighted and exponentially decayed matches, A_w denotes the weighted sum of misalignments between S_1 and S_2, and $C[\cdot]$ is a precomputed cumulative sum that integrates the exponential decay across the string length, serving as a normalization factor for the similarity score.

The ConvJW is implemented in Algorithm 5. For further calculation examples, see Example B.1.

5.3.4 Experiments

The effectiveness of the novel algorithms, ConvJ and ConvJW, was evaluated on a suite of datasets, segmented into two categories: Dataset A and Dataset B. The Dataset A is enumerated in Table 5.3.4, span various entities characterized by single attributes, as adapted from Cohen (2003) [130]. For Dataset B [229], outlined in Table 5.3.4, specific focus is placed on the first attribute, namely the name and title, for the experiments. These datasets include complex records from the Abt-Buy, Amazon-Google Products and DBLP-ACM sources, which are designed for benchmarking entity resolution tasks. The datasets are part of a collection curated by the DBS Uni Leipzig, aimed at facilitating research in the domain of entity resolution by providing a diverse set of scenarios where entities need to be matched across different data sources despite discrepancies in their representations [230].

Dataset	#Number of strings	Dataset	#Number of strings
Animal	5,709	Game	911
Bird Kunkel	336	Park	654
Bird Nybird	982	Restaurant	863
Bird Scott1	38	Ucd-people	90
Bird Scott2	719	Census	841
Business	2,139		

Table 5.1: Dataset A used in experiments from original sources [130]

Dataset	#Tuples	#True matches	#Attributes
Abt-Buy	1081-1092	1097	4
Amazon-GoogleProducts	1363-3226	1300	4
DBLP-ACM	2614-2294	2224	4

Table 5.2: Dataset B used in experiments from original sources [230]

This analysis leveraged the non-interpolated average precision, adopting methodologies from prior works [130], [178]. Precision and recall are defined as follows:

$$\text{Precision} = \frac{c(i)}{i}, \tag{5.27}$$

$$\text{Recall} = \frac{c(i)}{m}, \tag{5.28}$$

where $c(i)$ represents the count of correct matches up to rank i, and m denotes the total correct matches. The interpolated precision at recall level r maximizes precision for all ranks i satisfying $c(i)/m \geq r$. Performance comparison among similarity functions utilizes the maximum F1-score, calculated by:

$$\text{F1-score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \times 100\%. \tag{5.29}$$

Results are depicted in Tables 1 and 2 for Dataset A and B, emphasizing the comparative analysis of current state-of-the-art character-based similarity functions.

In the comprehensive analysis, the Convolutional Jaro (ConvJ) and Convolutional Jaro-Winkler (ConvJW) metrics were assessed across multiple datasets to evaluate their performance in comparison with state-of-the-art character based approximate string matching such as Jaro [190], Jaro-Winkler [191], Levenshtein [123], Damerau-Levenshtein [192], Smith-Waterman [70], and Needleman-Wunsch [69]. The evaluation focused on the F1-score, a critical metric reflecting both precision and recall, to provide a balanced view of

each algorithm's accuracy and efficiency.

Superior F1-Score Performance: The results, particularly highlighted in Tables 5.3.4 and 5.3.4, demonstrate the superior performance of ConvJ and ConvJW algorithms. ConvJW, with a σ setting of 2, achieved the highest F1-score of 86.32% on Dataset A, surpassing the traditional Jaro-Winkler score of 81.45%. This improvement emphasizes the efficacy of integrating a convolutional methodology and Gaussian weighting to accurately capture the proximity of character positions with improved precision. Similarly, on Dataset B, ConvJ with a σ setting of 0.5 achieved an F1-score of 89.13%, outperforming Jaro-Winkler's 86.17%, further affirming the robustness of our proposed metrics in varied dataset contexts.

Optimal Sigma (σ) Settings: The performance sensitivity to the σ parameter was evident, where ConvJand ConvJW's effectiveness varied with changes in σ. For instance, ConvJ's performance peaked at a σ value of 0.5 on Dataset B, indicating the importance of tuning this parameter to balance between emphasizing close character matches and accommodating positional variances. This adaptability allows for fine-tuning the algorithms to match the specific requirements of different datasets, contributing significantly to their superior performance (Figure 5.4 and Figure 5.5).

Similarity Function	F1-score
ConvJW($\sigma = 2, \beta = 0.1$)	87.95 %
ConvJW($\sigma = 1, \beta = 0.1$)	87.81 %
ConvJW($\sigma = 0.5, \beta = 0.1$)	87.14 %
ConvJ($\sigma = 1$)	84.99 %
ConvJ($\sigma = 2$)	84.72 %
ConvJ($\sigma = 0.5$)	84.54 %
Jaro-Winkler	81.45 %
Damerau-Levenshtein	76.86 %
Levenshtein	76.83 %
Needleman-Wunsch	76.25 %
Smith-Waterman	75.71 %
Jaro	75.29 %

Table 5.3: F1-score Comparison of character-based similarity functions ranked in descending order of F1-score for Dataset A

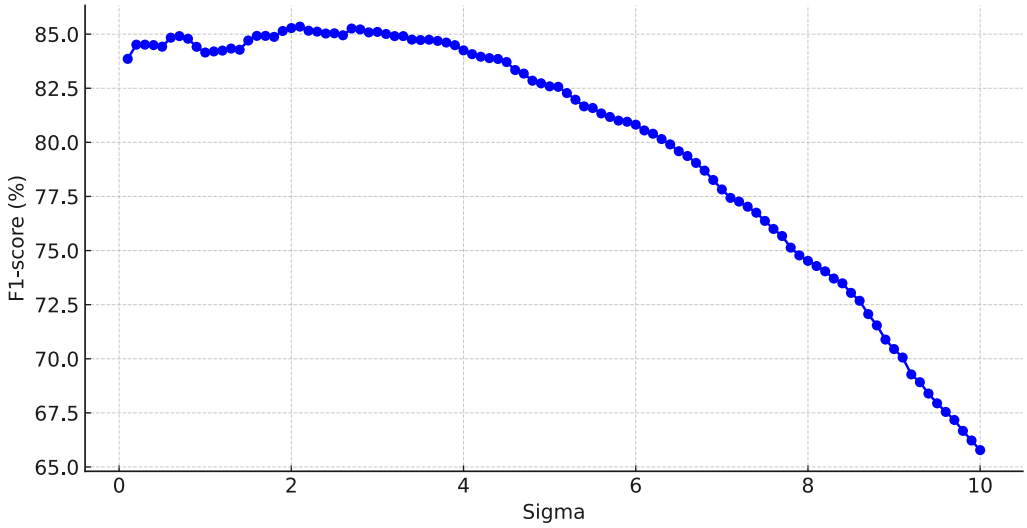

Figure 5.4: F1-score performance as a function of the σ parameter in the ConvJ algorithm. The graph demonstrates the optimal σ value for maximizing the F1-score, illustrating the algorithm's sensitivity to this parameter for Dataset A.

Similarity Function	F1-score
ConvJ($\sigma = 0.5$)	89.13%
ConvJ($\sigma = 1$)	89.03%
ConvJ($\sigma = 2$)	88.55%
ConvJW($\sigma = 0.5, \beta = 0.1$)	88.39%
ConvJW($\sigma = 1, \beta = 0.1$)	88.17%
ConvJW($\sigma = 2, \beta = 0.1$)	87.96%
JaroWinkler	86.17%
Needleman-Wunsch	85.85%
Jaro	85.78%
Levenshtein	85.39%
Damerau-Levenshtein	85.37%
Smith-Waterman	84.38%

Table 5.4: F1-score Comparison of character-based similarity functions ranked in descending order for Dataset B

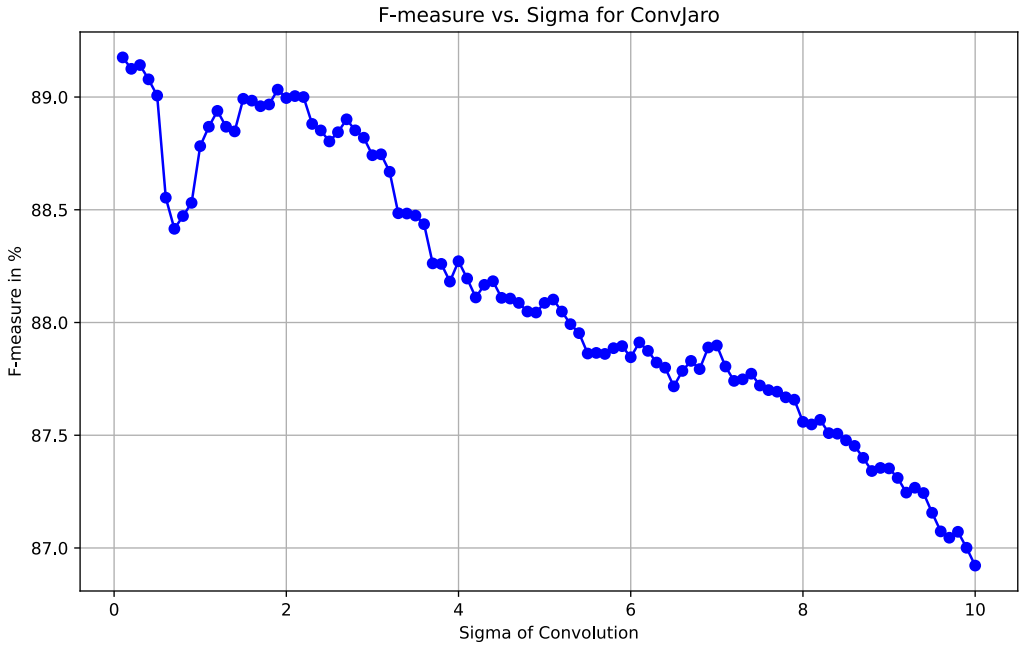

Figure 5.5: F1-score performance as a function of the σ parameter in the ConvJ algorithm. The graph demonstrates the optimal σ value for maximizing the F1-score, illustrating the algorithm's sensitivity to this parameter for Dataset B.

5.3.5 Time and Space Complexity

The time complexity of both the Jaro and Jaro-Winkler similarity algorithms is $\mathcal{O}(|S_1||S_2|)$, where $|S_1|$ and $|S_2|$ denote the lengths of the two strings being compared. This complexity arises from the requirement to potentially compare each character in one string to every character in the other string within a certain matching window, scaling with the product of the lengths of the two strings.

The memory complexity for these algorithms is $\mathcal{O}(|S_1| + |S_2|)$. This is due to the need to store intermediate match information, such as matched characters, for each string. This storage is necessary to calculate the final similarity score, including handling transpositions for the Jaro algorithm and prefix similarity adjustments in the Jaro-Winkler extension.

These complexities suggest that while Jaro and Jaro-Winkler are relatively efficient for shorter strings, their performance may degrade for longer strings due to the quadratic nature of their time complexity. Despite this, they are popular for many practical applications involving string similarity and approximate matching due to their simplicity and effectiveness.

The ConvJ algorithm introduces more efficient computational approach to approximate string similarity, characterized by a quasilinear time complexity with respect to the length of one string and the fixed window size. Specifically, the time complexity is denoted as $\mathcal{O}(|S_1|w)$, where $|S_1|$ is the length of the first string and w is the fixed window size utilized in the similarity calculation. This efficiency is achieved by limiting comparisons to a fixed window around each character, significantly reducing the number of operations compared to traditional quadratic approaches.

Memory complexity for ConvJ is primarily influenced by the storage of precomputed Gaussian weights and intermediate calculations. Given the fixed window size w, the memory requirement is $\mathcal{O}(w)$, accounting for the Gaussian weight array and temporary variables used during computation. This compact memory footprint makes ConvJ particularly suitable for applications with stringent memory constraints.

Performance assessments were conducted on an Intel i7 11370H processor with 16GB RAM, comparing execution times and performance deltas to gauge the algorithms' efficiency. The summarized outcomes in Tables 5.3.5 and 5.3.5 for Dataset A and B reveal the computational advantages of ConvJ and ConvJW against established string matching algorithms.

Computational Efficiency: Notably, ConvJ and ConvJW not only excelled in accuracy but also in computational efficiency. As evidenced in Tables 5.3.5 and 5.3.5, ConvJW($\sigma = 0.5$) recorded the fastest execution time on Dataset B with 0:10:158, showcasing a substantial speed advantage over traditional metrics like Jaro (1:21:675) and Jaro-Winkler (1:33:368). This efficiency is paramount for large-scale applications, enabling rapid and precise string similarity assessments across extensive datasets.

Algorithm	Time (mm:ss:ms)	Performance Δ (%)
ConvJ($\sigma = 0.5$)	0:02:412	0.00%
ConvJW($\sigma = 0.5$)	0:02:660	+10.28%
ConvJ($\sigma = 1$)	0:03:607	+49.54%
ConvJW($\sigma = 1$)	0:03:841	+59.25%
ConvJ($\sigma = 2$)	0:05:540	+129.68%
ConvJW($\sigma = 2$)	0:05:577	+131.22%
Jaro	0:10:314	+327.61%
Jaro-Winkler	0:10:736	+345.11%
Levenshtein	0:33:629	+1294.24%
Damerau-Levenshtein	0:58:098	+2308.71%
Needleman-Wunsch	1:13:331	+2940.26%
Smith-Waterman	1:24:695	+3411.40%

Table 5.5: Time Performance of Character-Based Similarity for Dataset A

Algorithm	Time (mm:ss:ms)	Performance Δ (%)
ConvJ($\sigma = 0.5$)	0:10:158	0.00%
ConvJW($\sigma = 0.5$)	0:12:733	+25.34%
ConvJ($\sigma = 1$)	0:15:948	+56.94%
ConvJW($\sigma = 1$)	0:17:424	+71.54%
ConvJ($\sigma = 2$)	0:25:537	+151.49%
ConvJW($\sigma = 2$)	0:27:048	+166.32%
Jaro	1:21:675	+705.88%
Jaro-Winkler	1:33:368	+819.36%
Levenshtein	6:20:035	+3759.22%
Damerau-Levenshtein	11:20:111	+6708.99%
Needleman-Wunsch	15:45:774	+9296.85%
Smith-Waterman	16:38:783	+9815.41%

Table 5.6: Time Performance of Character-Based Similarity for Dataset B

5.4 Fuzzy Record Similarity (FRS)

The proposed model is based on maximum weight matching in a bipartite graph [203], [205], [206] that satisfies an axiom (S1) for symmetry mentioned above. The exact optimal solution to the combinatorial assignment problem can be solved by the Kuhn–Munkres algorithm [231]–[233].

A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a set $\mathcal{V} = \mathcal{R}_1 \cup \mathcal{R}_2$ of vertices (the tokens of both records) and a set E of pairs of vertices, called edges. For an edge $e = (X_i, Y_j)$, it is stated that the endpoints of e are X_i and Y_j; it is also stated that e is incident to X_i and Y_j.

A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is bipartite if the vertex set $\mathcal{V}$ can be partitioned into two sets $\mathcal{R}_1$ and $\mathcal{R}_2$ (the bipartition) such that $\mathcal{R}_1 \cap \mathcal{R}_2 = \emptyset$ and no edge in E has both endpoints in the same set of the bipartition.

A matching $\mathcal{M} \subseteq \mathcal{E}$ is a collection of edges such that every vertex is incident to at most one edge of $\mathcal{M}$. If a vertex has no edge incident to it, then the vertex is said to be exposed (or unmatched). A matching is perfect if no vertex is exposed; in other words, a matching is perfect if its cardinality is equal to $|\mathcal{R}_1| = |\mathcal{R}_2|$.

Theorem 44 (König)**.** *For any bipartite graph, the maximum size of a matching is equal to the minimum size of a vertex cover.*

This theorem is an expression of the equality of the primary and dual problem in linear programming. The Kuhn–Munkres algorithm is based on such dual tasks.

Definition 29 (Maximum Weight Matching Problem)**.** *Given a weight $c_{i,j}$ for all $(i, j) \in \mathcal{E}$. Given a matching, let its incidence vector be $\mathbf{x}$ where $x_{i,j} = 1$ if $(i, j) \in \mathcal{M}$ and 0 otherwise. One can formulate the maximum weight matching problem as follows: its*

objective function is

$$x^* = \arg\max \sum_{i=1}^{|\mathcal{R}_1|} \sum_{j=1}^{|\mathcal{R}_2|} c_{i,j} x_{i,j} = \arg\max c^T x, \tag{5.30}$$

subject to

$$\sum_{i=1}^{|\mathcal{R}_1|} x_{i,j} = 1, \tag{5.31}$$

$$\sum_{j=1}^{|\mathcal{R}_2|} x_{i,j} = 1. \tag{5.32}$$

The original algorithm has a polynomial time complexity $\mathcal{O}(|\mathcal{V}|^4)$ but it has been shown that it can be modified to achieve an $\mathcal{O}(|\mathcal{V}|^3)$ running time. Note that the best known time complexity for the maximum weighted matching is $\mathcal{O}(|\mathcal{V}||\mathcal{E}| + |\mathcal{V}|^2 \log |\mathcal{V}|)$ [234], [235]. The class of these combinatorial optimization problems belongs to the class of *NP-complete* problems.

For the calculation of textual similarity, the normalized edit similarity in the range $[0, 1]$, already defined in Theorem (45), is utilized

$$c_{i,j} = s_n(X_i, Y_j). \tag{5.33}$$

Let's mention that several normalization factors have been developed [111], [196], [197] and most of them violate the triangle inequality as a condition of being a similarity. However, in [111] there has been a normalized similarity, and this has been further generalized into similarity space [OR-1]. On the other hand, it also has been found that such normalization factors don't affect the accuracy, with fluctuations in the test results $\approx 1\%$. This accuracy could be considered as a statistically acceptable error.

A normalization of $d(X_i, Y_j)$ differences, commonly referred to as the edit distance, for Levenshtein similarity is now introduced in the interval $[0, 1]$, a widely used approach in [178], [189], [200], [205],

$$s(X_i, Y_j) = 1 - \frac{d(X_i, Y_j)}{\max\{|X_i|, |Y_j|\}}. \tag{5.34}$$

It can be easily shown, for example 5.4, that this normalization also violates the triangle inequality, and therefore all the distance or similarity functions introduced in the articles based on this normalization are not metrics and cannot be well-used in applications like cluster analysis.

Example. *Considering the strings* $X =$ "*ab*", $Y =$ "*abc*", *and* $Z =$ "*bc*", *the following is*

obtained:

$$\begin{aligned} &d(X,Z) \leq d(X,Y) + d(Y,Z), \\ &\frac{d(X,Z)}{max\{|X|,|Z|\}} \leq \frac{d(X,Y)}{max\{|X|,|Y|\}} + \frac{d(Y,Z)}{max\{|Y|,|Z|\}}, \\ &\frac{2}{2} \not\leq \frac{1}{3} + \frac{1}{3}. \end{aligned} \tag{5.35}$$

Theorem 45 (Normalized Edit similarity)**.** *Let Σ be a finite alphabet, and let Σ^* denote the set of all strings over Σ. Given $X, Y \in \Sigma^*$, the Generalized Rozinek Similarity over Σ is the normalized edit similarity*

$$s_n(X,Y) = \frac{s(X,X) + s(Y,Y) - d(X,Y)}{s(X,X) + s(Y,Y) + d(X,Y)} = \frac{|X| + |Y| - d(X,Y)}{|X| + |Y| + d(X,Y)}, \tag{5.36}$$

where $s_n(X,Y)\colon \Sigma \times \Sigma^* \to [0,1] \subset \mathbb{R}$ and $|\,.\,|$ denotes the cardinality of a set, specifically the number of characters of the string.

Proof. Without loss of generality, it may be assumed that a self-similarity $s(X,X)$ is a set function, particularly that $s(X,X)$ is a measure $\mu(X)$ on a σ-algebra on the finite alphabet Σ. The set of all strings Σ^* induces a σ-algebra on the finite alphabet Σ, resulting in a measure space represented as the triple (Σ, Σ^*, μ).

In fact, in the abstract meaning, the edit distance $d(x,y)$ is a symmetric difference of ordered sets, where $d(X,Y) = \mu(X \triangle Y)$ is calculated by an algorithm of dynamic programming. For finite sets, the cardinality is a natural measure of size. In this case, the cardinality $|\,.\,|$ of sets is defined as the number of characters in the sets. It is expressed

$$\begin{aligned} s_R(X,Y) = s_n(X,Y) &= \frac{s(X,X) + s(Y,Y) - d(X,Y)}{s(X,X) + s(Y,Y) + d(X,Y)} \\ &= \frac{\mu(X) + \mu(Y) - \mu(X \triangle Y)}{\mu(X) + \mu(Y) + \mu(X \triangle Y)} \\ &= \frac{|X| + |Y| - d(X,Y)}{|X| + |Y| + d(X,Y)}. \end{aligned} \tag{5.37}$$

Since it is proven from Theorem 18 that $s_R(X,Y)$ is a normalized similarity, the proof is complete. Alternatively, a second proof [OR-1], [111] is provided to support the statement

$$\frac{|X| + |Y| - d(X,Y)}{|X| + |Y| + d(X,Y)} = 1 - \frac{2d(X,Y)}{|X| + |Y| + d(X,Y)} = 1 - d_n(X,Y), \tag{5.38}$$

where $d_n(x, y)$ is a normalized edit distance of the form

$$d_n(X, Y) = \frac{2d(X, Y)}{|X| + |Y| + d(X, Y)}. \tag{5.39}$$

In [111] it is further proved that d_n is a normalized distance metric. Hence by the duality between normalized similarity and normalized distance metrics, $s_n(x, y) = 1 - d_n(x, y)$ is also proven [OR-1]. □

Now, a scenario is considered in which the prediction of the edit distance is desired based solely on the lengths of the strings X and Y. The calculation of the edit distance $d(X, Y)$ itself is computationally expensive, in quadratic time $\mathcal{O}(|X||Y|)$, So, a rough estimate of the worst-case expected edit distance for the minimum edit normalized similarity, given by the threshold $s_n(X, Y) \geq \alpha$, is desired. An expected edit distance $\sup_\alpha d(X, Y) = d(\alpha, |X|, |Y|)$ is introduced, depending on already known parameters – the threshold α and the lengths of the strings $|X|$ and $|Y|$. Finally, a prediction is obtained, which is computationally feasible in constant time $\mathcal{O}(1)$ and will assist in developing an optimal filter in subsequent chapters.

Definition 30 (Expected Edit Distance)**.** *Let the worst case expected edit distance be* $d(\alpha, |X|, |Y|) \colon \mathbb{R} \times \mathbb{N}^+ \times \mathbb{N}^+ \to \mathbb{N}_0$ *the maximal possible edit distance* $d(X, Y) \colon \Sigma^* \times \Sigma^* \to \mathbb{N}_0$ *for string lengths* $|X|$ *and* $|Y|$ *and a similarity given by fixed* $\alpha \in [0, 1]$*.*

Theorem 46 (Threshold of Normalized Edit similarity)**.** *Let the edit distance be* $d(X, Y)$*, the worst case of the expected distance function be* $d(\alpha, |X|, |Y|)$*, and write the floor function by* $\lfloor . \rfloor$*. Then*

$$s_n(X, Y) \geq \alpha \iff d(X, Y) \leq d(\alpha, |X|, |Y|) = \left\lfloor \frac{1 - \alpha}{1 + \alpha}(|X| + |Y|) \right\rfloor, \tag{5.40}$$

where $\alpha \in [0, 1] \subset \mathbb{R}$ *is a threshold of the normalized similarity given by* $s_n(X, Y) \geq \alpha$*.*

Proof. According to Theorem 20 and substituting for self-similarities which equal the corresponding cardinality of the sets, $s(X, X) = |X|$, $s(Y, Y) = |Y|$, the expected distance $d(\alpha, |X|, |Y|)$ reaches a maximum just when the similarity is minimal under the lowest similarity given by the threshold α. This occurs if and only if $s_n(X, Y) = \alpha$ is substituted:

$$\begin{aligned} d(X, Y) = \lfloor d_R \rfloor &= \left\lfloor \frac{1 - s_n(X, Y)}{1 + s_n(X, Y)}(s(X, X) + s(Y, Y)) \right\rfloor \\ &\leq \sup_\alpha d(X, Y) = \left\lfloor \frac{1 - \alpha}{1 + \alpha}(|X| + |Y|) \right\rfloor = d(\alpha, |X|, |Y|). \end{aligned} \tag{5.41}$$

Since the edit distance is an integer, the floor function is used to remove any undefined fractional part. □

Definition 31 (Fuzzy Record Similarity - FRS)**.** *Let M be a matching collection of edges connecting pairs of tokens, and the cardinality $|\mathcal{M}| = \min\{|\mathcal{R}_1|, |\mathcal{R}_2|\}$, especially for cases of non-perfect matching, $|\mathcal{R}_1| \neq |\mathcal{R}_2|$. An expected value of the similarity is defined as an average of the normalized edit similarity between $\mathcal{R}_1$ and $\mathcal{R}_2$ as follows:*

$$\mathbb{E}[s_n(\mathcal{R}_1, \mathcal{R}_2)] = s_n(\mathcal{R}_1, \mathcal{R}_2) = \frac{\sum_{(i,j)\in\mathcal{M}} s_n(X_i, Y_j)}{|\mathcal{M}|}. \tag{5.42}$$

Theorem 47. *The Fuzzy Record similarity (FRS) is a normalized similarity.*

Proof. FRS satisfies (N1) because the maximum weight matching in a bipartite graph is a symmetric measure. Continuing from Definition 31, Theorem 45 is applied along with convex combinations in Theorem 5, and then the remaining axioms are easily proven. □

Corollary 6 (Identity FRS and Fuzzy Overlap Similarity)**.** *The Fuzzy Record similarity equals the Fuzzy Overlap Similarity if and only if $\delta = 0$:*

$$s_n(\mathcal{R}_1, \mathcal{R}_2) = sim_O(\mathcal{R}_1, \mathcal{R}_2). \tag{5.43}$$

Proof. It can be expressed directly

$$\begin{aligned} s_n(\mathcal{R}_1, \mathcal{R}_2) &= \frac{\sum_{(i,j)\in\mathcal{M}} s_n(X_i, Y_j)}{|\mathcal{M}|} = \frac{\sum_{(i,j)\in\mathcal{M}} s_n(X_i, Y_j)}{\min\{|\mathcal{R}_1|, |\mathcal{R}_2|\}} \\ &= \frac{|\mathcal{R}_1 \mathbin{\widetilde{\cap}}_{\delta=0} \mathcal{R}_2|}{\min\{|\mathcal{R}_1|, |\mathcal{R}_2|\}} = \mathrm{sim}_O(\mathcal{R}_1, \mathcal{R}_2). \end{aligned} \tag{5.44}$$

whenever a second threshold is removed by setting $\delta = 0$, a complete bipartite graph is obtained, therefore $\sum_{(i,j)\in\mathcal{M}} s_n(X_i, Y_j) = |\mathcal{R}_1 \mathbin{\widetilde{\cap}}_{\delta=0} \mathcal{R}_2|$. □

In contrast to the Fuzzy Jaccard Similarity from Definition 24, a well-designed generalized similarity suitable for text mining and cluster analysis is offered as the first introduced advanced fuzzy record similarity technique.

Graphically, the whole procedure is shown in Figure 5.6 and 5.7.

For the time complexity, note that in most cases in large databases, in each attribute there are stored short strings with few tokens, hence the computation is well feasible and fast in many real-world scenarios. However in large storage with millions of records, such a time complexity of the algorithm is not usable in real-time approximate string matching and search. There could be developed a filter, also called a blocking technique, using a method with much lower time complexity and thus filter many records and significantly

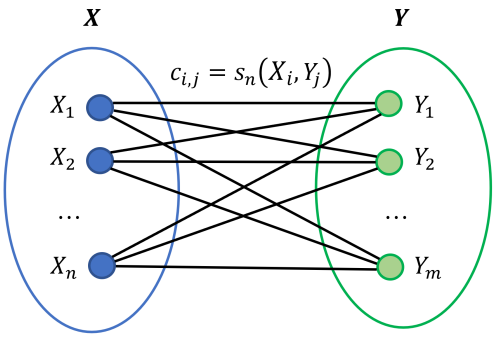

Figure 5.6: The construction of a complete bipartite graph where every token vertex of the first record set $\mathcal{R}_1$ is connected to every token vertex of the second record set $\mathcal{R}_2$.

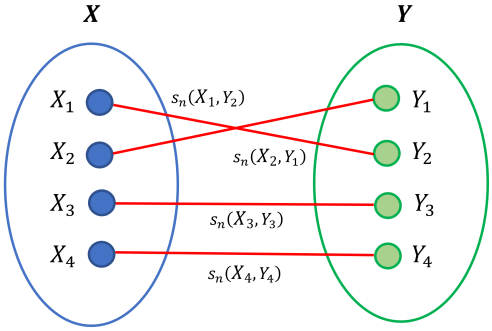

Figure 5.7: Maximum weighted bipartite matching of two records $\mathcal{R}_1$ and $\mathcal{R}_2$ and adjacent edges of token pairs X_i, Y_j weighted by normalized similarity $s_n(X_i, Y_j)$.

reduce the large comparison space [177], [199], [200]. Practically all used methods are sub-optimal filters which during filtering lose true positive candidates. The reason for this observation is that these methods are not explicitly mathematically united.

The intention of this thesis is to show in general that it is possible to derive an optimal filter that is less time-consuming and serves as the lower bound of the threshold related to a more time-consuming algorithm of approximate string matching. In this case, a two-stage method is proposed, where a less time-consuming filter, such as a Q-gram filter, could be used for the proposed token-based model FRS with polynomial time complexity. The Q-gram filter is also one of the most used indexing methods for search engines in real-world applications. Thresholding by least shared Q-grams is known as T-overlap similarity join [200]. In the next chapters, a new optimal Q-gram filter for FRS will be described, the purpose of which is to solve the T-overlap similarity join problem.

5.5 Count Q-gram Filter

Some basic types of Q-gram filtering are distinguished, including count filtering, positional filtering, prefix filtering, and length filtering [177], [236]. Two of these types will be described in this thesis, providing a basic description of their ideas:

Count Filtering

The intuition behind count filtering is that strings with greater edit similarity than a threshold $s_n \geq \alpha$ have a large number of Q-grams in common, based on Theorem 48, Corollary 8 and 49.

At the beginning of this chapter, the concept of Q-gram similarity lying in similarity space will be introduced.

Definition 32 (Q-gram Similarity)**.** *Let Σ be a finite alphabet, and let Σ^* denote the set of all strings over Σ. The function $s_q\colon \Sigma^* \times \Sigma^* \to \mathbb{N}_0$ is the Q-gram similarity function for the strings $\mathcal{R}_1$, Y*

$$s_q(Q_X, Q_Y) = |Q_X \cap Q_Y|, \tag{5.45}$$

where Q_X and Q_Y are the corresponding Q-gram sets.

Corollary 7. *Q-gram Similarity $s_q(Q_X, Q_Y)$ is a similarity.*

Proof. It has been already shown in [OR-1] that a set intersection is a elementary similarity satisfying (S1), (S2), (S3) and (S4) from Definition 5. □

A normalized similarity called the Jaccard metric is obtained by dividing by $|Q_X \cup Q_Y|$ [OR-1]. For further derivation, only a simple intersection is considered as a measurement of the Q-grams common to $\mathcal{R}_1$ and Y.

In [122], there is proposed a relation between the edit distance and the Q-gram method. Suppose given a pattern string $\mathcal{R}_1$ with length $|X|$ and a text string Y with length $|Y|$. Then the following theorem is given in [237]:

Theorem 48 (Q-gram Count Filtering [122], [237])**.** *Let $\mathcal{R}_1$ and Y be strings with the edit distance $d(X, Y)$. Then, the Q-gram similarity $|Q_X \cap Q_Y|$ of the token $\mathcal{R}_1$ and Y is at least*

$$t = \inf_d\{|Q_X \cap Q_Y|\} = \max\{|X|, |Y|\} - q + 1 - qd(X, Y), \tag{5.46}$$

where t is a Q-gram similarity threshold with respect to $d(X, Y)$.

Proof. [122] □

This observation is crucial for a family of algorithms focusing on string similarity search and similarity join based on edit distance constraints with many real-world applications in data cleaning, search engines and integration, which extend traditional exact search and exact join operations in databases by tolerating errors and inconsistencies in the data. See [200].

One of the common problems is to find the threshold of least sharing Q-grams for an allowed edit distance with a fixed maximum distance d_{max} which can be set as a parameter by the user. Denote by $d \in [0, d_{max}]$ an interval range of allowed edit distance.

Corollary 8 (Infimum of Q-gram for Edit Distance)**.** *Let $d \in [0, d_{max}]$ be an edit distance. Then there exists a lower bound*

$$\inf_d\{|Q_X \cap Q_Y|\} = \max\{|X|, |Y|\} - q + 1 - q\min\{d_{sing}, d_{max}\}. \tag{5.47}$$

Proof. $\inf_d\{|Q_X \cap Q_Y|\} = \inf_{d=d_{max}}\{|Q_X \cap Q_Y|\}$; the term $\min\{d_{sing}, d_{max}\}$ treats filter underflow below 0. □

A singularity of a Q-gram filter d_{sing} occurs whenever this lower bound is less than or equal to zero, as will be explained further on.

Corollary 9 (Supremum of Q-gram for Edit Distance)**.** *Suppose given an edit distance $d \in [0, d_{max}]$. Then there exists an upper bound*

$$\sup_d\{|Q_X \cap Q_Y|\} = \max\{|X|, |Y|\} - q + 1. \tag{5.48}$$

Proof. Similarly, an upper bound is obtained by maximizing $|Q_X \cap Q_Y|$ and using $d = 0$, hence $\sup_d\{|Q_X \cap Q_Y|\} = \sup_{d=0}\{|Q_X \cap Q_Y|\}$. □

In this new definition, an expression using infimum and supremum on the Q-gram set in relation to the edit distance d was introduced for the first time. A similar kind of equation is also introduced in [122], [237]. For simplicity, from now on, the threshold is written as $t = \inf_d\{|Q_X \cap Q_Y|\}$.

Definition 33 (Q-gram Singularity)**.** *It is said that there is a Q-gram singularity if no common Q-gram is guaranteed, and so Q-gram filtering has no effect and*

$$t = \inf_d\{|Q_X \cap Q_Y|\} \leq 0. \tag{5.49}$$

Corollary 10 (Length Singularity of Q-gram Filter)**.** *Let $|X|_{sing}$ be the string length where $t = 0$ for an edit distance $d(X, Y) > 0$. It is called a length singularity of the Q-gram filter for edit distance*

$$|X|_{sing} = (d(X, Y) + 1)q - 1. \tag{5.50}$$

Proof. A positive edit distance $d > 0$ is assumed. Then

$$t = |X| - q + 1 - qd(X, Y), \tag{5.51}$$

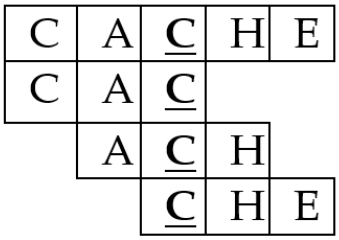

Figure 5.8: An example of a singularity of a trigram filter for $d = 1$ and $|X|_{sing} = 5$. Suppose the worst case for fixed $d = 1$ is the substitution in the position of character C destroying all trigrams $\{"CAC", "ACH", "CHE"\}$, hence $t = 0$

.

$$|X| = t + (d(X, Y) + 1)q - 1, \tag{5.52}$$

with $t = 0$ is proven. □

There is an example in Fig. 5.8.

5.5.1 Optimal Count Q-gram Filter for Character-Based Similarity

The normalization of the edit distance has received less attention in many scientific articles, but it is of great importance for many real-world applications that require measuring normalized similarity independently of the length of the string. The edit distance $d = 1$ on a token with length $|X| = 24$ is very often totally different from that of a token Y with, e.g. $|Y| = 6$. The normalization factor brings is well related to the human perception of the similarity of different objects. From similarity space theory, the tokens could often have different self-similarities. Self-similarity could be a measure of the number of extracted Q-grams or token lengths. For instance, having a German word X = 'Einkommensteuererklärung' (income tax return) and Y = 'Steuer' (tax), then $s(X, X) \geq s(Y, Y)$ when counting string lengths, common characters or Q-grams. Whenever two objects with different feature sizes are compared, it is questioned whether these objects are of different importance. In this model, it is asserted that they are not, leading to the following argument for normalization among tokens.

Similar to the previous equation (8), the normalized least sharing Q-gram pairs are reformulated as a threshold similarity as follows:

$$t_\alpha = \inf_\alpha \{|Q_X \cap Q_Y|\} = \max\{|X|, |Y|\} - q + 1 - qd(\alpha, |X|, |Y|), \tag{5.53}$$

where the edit distance is substituted with the worst-case scenario $d(X,Y) = d(\alpha, |X|, |Y|)$, which is utilized later in this thesis.

The occurrence of a singularity of the Q-gram filter t_α depends on the string lengths $|X|$ and $|Y|$ and the user-chosen threshold parameter $\alpha \in [0,1]$.

Corollary 11. — *Let $|X|$ and $|Y|$ be string lengths where $t_\alpha = 0$ for a normalized edit similarity $s_n(X,Y) \geq \alpha$. This is termed a singularity of the Q-gram filter*

$$|X|_{sing} = qd(\alpha, |X|, |Y|) + q - 1. \tag{5.54}$$

Proof. With $t_\alpha = 0$ it can be proven

$$\begin{aligned} 0 &= |X| - q + 1 - qd(\alpha, |X|, |Y|), \\ |X|_{sing} &= qd(\alpha, |X|, |Y|) + q - 1. \end{aligned} \tag{5.55}$$

For simplicity, it is assumed that the tokens have the same lengths $|X| = |Y|$. For illustration purposes, these examples are provided: if $d(\alpha, |X|, |Y|) = 1$, then $|X|_{sing} = |Y|_{sing} = 2q - 1$ for a trigram, where $|X|_{sing} = 5$, and for a bigram, where $|X|_{sing} = 3$.

For another example, if $d(\alpha, |X|, |Y|) = 2$, $|X|_{sing} = |Y|_{sing} = 3q - 1$ for the trigram $|X|_{sing} = 8$ and for the bigram $|X|_{sing} = 5$. □

Corollary 12 (Edit Distance Singularity of Q-gram Filter)**.** *Suppose given an expected edit distance $d_{sing}(\alpha, |X|, |Y|)$ which is a singularity of the Q-gram filter for an edit distance, that is, where $t_\alpha = 0$. Then,*

$$d_{sing}(\alpha, |X|, |Y|) = \left\lceil \frac{|Q_X|}{q} \right\rceil \tag{5.56}$$

where Q_X is the Q-gram set of string $\mathcal{R}_1$.

Proof. Since $t_\alpha = 0$, then, similar to before,

$$\begin{aligned} 0 &= \max\{|X|, |Y|\} - q + 1 - qd_{sing}(\alpha, |X|, |Y|), \\ d_{sing}(\alpha, |X|, |Y|) &= \left\lceil \frac{\max\{|X|, |Y|\} - q + 1}{q} \right\rceil, \\ d_{sing}(\alpha, |X|, |Y|) &= \left\lceil \frac{\sup\{|Q_X \cap Q_Y|\}}{q} \right\rceil. \end{aligned} \tag{5.57}$$

For simplicity, again assume $|X| = |Y|$. Finally, it is obtained

$$d_{sing}(\alpha, |X|, |Y|) = \left\lceil \frac{|Q_X|}{q} \right\rceil. \tag{5.58}$$

The fractional part means the remaining Q-grams which have the size $|Q_X| \bmod q$. Additionally, the remaining Q-grams are destroyed with $d = 1$, thus the ceiling function $\lceil . \rceil$ is used to handle the fractional part. □

Simply explained, a Q-gram filter can work efficiently if $d(X, Y) \leq d_{sing}(\alpha, |X|, |Y|)$. If equality holds, all Q-grams could be destroyed, as the worst case.

Corollary 13. *(Edit Similarity Singularity of Q-gram Filter) Let α_{sing} be a singularity threshold of a Q-gram filter where $t_\alpha = 0$.*

$$\alpha_{sing} = \frac{2q|X| - |X| + q - 1}{2q|X| + |X| - q + 1} = \frac{2q|X| - |Q_X|}{2q|X| + |Q_X|}. \tag{5.59}$$

Proof. Assuming an unfractional part, it is obtained that

$$\begin{aligned} d(\alpha, |X|, |Y|) &= d_{sing}(\alpha, |X|, |Y|), \\ \frac{1-\alpha}{1+\alpha}(|X| + |Y|) &= \frac{\sup\{|Q_X \cap Q_Y|\}}{q}, \\ \frac{1-\alpha}{1+\alpha} &= \frac{\sup\{|Q_X \cap Q_Y|\}}{q(|X| + |Y|)}, \\ \alpha &= \frac{q(|X| + |Y|) - \sup\{|Q_X \cap Q_Y|\}}{q(|X| + |Y|) + \sup\{|Q_X \cap Q_Y|\}}. \end{aligned} \tag{5.60}$$

This is proven by finding a root of the equation for α. Assuming the strings have equal lengths, this can be simplified to

$$\alpha = \frac{2q|X| - |Q_X|}{2q|X| + |Q_X|} = \frac{2q|X| - |X| + q - 1}{2q|X| + |X| - q + 1}. \tag{5.61}$$

□

In other words, a threshold $\alpha \geq \alpha_{sing}$ should always be selected for corresponding token lengths to ensure the effectiveness and efficiency of the Q-gram filter in filtering any dissimilar records.

Corollary 14. *Shorter strings less than $|X| < \left\lceil \frac{1+\alpha}{2-2\alpha} \right\rceil$ must have an exact match with size for all Q-grams $|Q_X|$ for the Q-gram filter*

$$|X| < \left\lceil \frac{1+\alpha}{2-2\alpha} \right\rceil \implies t_\alpha = |Q_X| = |X| - q + 1. \tag{5.62}$$

Proof. Thus, it is shown that a non-integer fractional part $d(\alpha, |X|, |Y|) < 1$ leads to an exact match after applying the floor function, i.e., $d(\alpha, |X|, |Y|) < 1 \implies \lfloor d(\alpha, |X|, |Y|) \rfloor =$

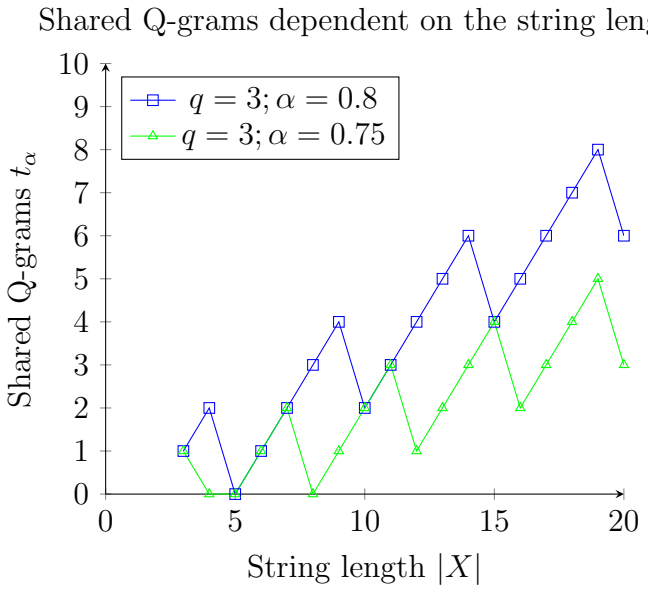

Figure 5.9: Sawtooth function of different string lengths $|X|$ and fixed q and α. The zero crossings of t_α illustrate singularities of the Q-gram filter. Similar to results in [238].

0.

$$\left\lfloor \frac{2|X| - 2\alpha|X|}{\alpha + 1} \right\rfloor < 1, \qquad |X| < \left\lceil \frac{\alpha + 1}{2 - 2\alpha} \right\rceil. \tag{5.63}$$

□

Indeed, for $d = 1$ and $\alpha = 0.8$, $|X| = 5$ is obtained. Shorter strings must always have $s_n(X, Y) < \alpha$ for $d > 0$.

A singularity of a Q-gram filter could cause performance problems on certain string lengths $|X|_{sing}$ due to generating a large candidate set. To resolve this kind of problem, this model should be extended with at least 1 padding Q-gram, denoted by $p \in \Sigma^*$. From nature, empirical observation, and some blocking techniques [177], [199], [200], increasing the importance of initial letters seems reasonable. For an initial solution, a prefix or postfix '$p = \#$' or both could be defined for each token, and the Q-gram sets could be extended by about $|Q_X| + |p|$ Q-grams.

Corollary 15. *A smooth Q-gram set is a set of Q-grams extended by padding the total length $|Q_X| + |p|$ and with no singularity for $t_\alpha = 0$ for a normalized edit similarity $s_n(X, Y) \geq \alpha$ and an expected edit distance $d(\alpha, |X|, |Y|) > 0$. There is always at least*

some shared Q-grams, equal to the shift of the padded characters $|p|$.

$$t_{\alpha_p} = t_\alpha + |p| \tag{5.64}$$

Proof. Let p be the (finite) number of padded characters. Then

$$\begin{aligned} t_{\alpha_p} &= \max\{|X+p|, |Y+p|\} - q + 1 - qd(\alpha, |X|, |Y|) \\ &= \max\{|X|, |Y|\} + |p| - q + 1 - qd(\alpha, |X|, |Y|) \\ &= t_\alpha + |p|, \end{aligned} \tag{5.65}$$

where $t_\alpha = 0$. Note that padding characters p are not used to calculate the expected editing distance $d(\alpha, |X|, |Y|)$. □

The information about a singularity of a Q-gram filter or the near neighborhood around a singularity plays an important role. Many articles have compared, on different data sets, the use of padding Q-grams, reaching the same conclusion: this increases the F-measure statistics [239], [240]. A theoretical explanation for why this happens has never been seen in any framework. It is believed that singularity provides an explanation for this specific behavior. The padding characters have a "smoothing" effect on the token boundaries, which is why the result is called a "smooth Q-gram set". In another interpretation, more information for classification is obtained if the feature set is extended.

Another solution could be to combine feature extraction with multiple sizes of the Q-grams, e.g. unigrams, bigrams, and trigrams. The singularities of those Q-gram filters would be mutually resolved. A generalization of such a combination could be expressed as a sum of shared Q-grams over different Q-gram sizes t_1, t_2, t_3 and their constants $q_1 = 1$, $q_2 = 2$, $q_3 = 3$.

$$\begin{aligned} &t_{1,2,3} = t_1 + t_2 + t_3 = \\ &3\max\{|X|, |Y|\} - q_1 - q_2 - q_3 + 3 - d(\alpha, |X|, |Y|)(q_1 + q_2 + q_3) \end{aligned} \tag{5.66}$$

and evaluating

$$t_1 + t_2 + t_3 = 3\max\{|X|, |Y|\} - 3 - 6d(\alpha, |X|, |Y|). \tag{5.67}$$

5.5.2 Optimal Count Q-gram Filter for Token-Based Similarity

Set similarity join identifies all pairs of sets within a single record collection or across two different collections when the similarity score is above a certain threshold α [177], [179].

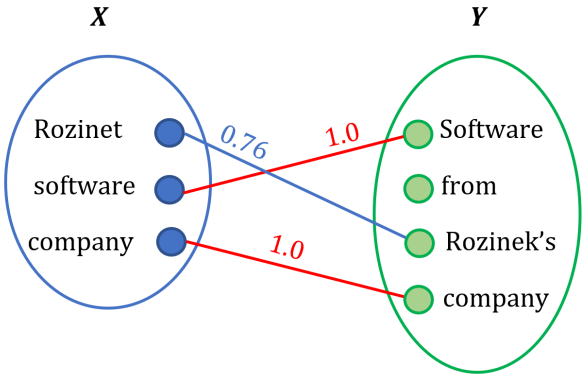

Figure 5.10: Maximum weighted bipartite matching of two records $\mathcal{R}_1$ and $\mathcal{R}_2$.

This process is an essential operation in many applications, such as data cleaning and integration, personalized recommendation and record deduplication.

As shown in the Fig. 5.12, the focus is on a two-step method for similarity join: firstly, using a count Q-gram filter, and secondly, employing maximum weighted bipartite matching as the best approach for solving the combinatorial assignment problem [204]. This method involves fuzzy token similarity in bipartite matchings, recognized state-of-the-arts for its high accuracy in classifying records as matches or non-matches in an error-tolerant manner. However, this approach results in polynomial time complexity, $\mathcal{O}(n^3)$, which is managed by the Kuhn-Munkres algorithm (Fig. 5.10).

To improve the efficiency of similarity joins, exhaustive pair comparisons are avoided by applying a count Q-gram filter. In real-world applications, Ukkonen's lemma [198], [237] is used to establish a direct relation between Q-gram similarity and edit distance (Levenshtein), which acts as a rigorous mathematical filter to reduce the number of comparisons without missing any true matches. As indicated in the figure, the Q-gram filter operates in constant time, $\mathcal{O}(1)$ on a pre-built inverted Q-gram index. This significant reduction in comparison pairs enables real-time processing, as only a small fraction of candidate pairs is further verified for actual matches in the second stage of Fuzzy Bipartite Matching (Fig. 5.12).

According to Ukkonen's lemma [198], [237], let X and Y be tokens with the edit distance $d(X, Y)$. Then, the Q-gram similarity $|Q_X \cap Q_Y|$ of the tokens X and Y is at least $t = \inf_d\{|Q_X \cap Q_Y|\} = \max\{|X|, |Y|\} - q + 1 - qd(X, Y)$, where t is a Q-gram similarity threshold with respect to $d(X, Y)$ and q is the Q-gram length.

The problem often arises with the assumption that this constraint is applied to the entire record [237]. Consider two records $\mathcal{R}_1$ and $\mathcal{R}_2$, split into sets of tokens $X_i \in \mathcal{R}_1$ and $Y_j \in \mathcal{R}_2$. This simplification is demonstrated in the example below, leading to differing results of the Q-gram filter as a constraint of fuzzy bipartite matching. This gap in this thesis's goal is addressed.

Example. *Consider two records $\mathcal{R}_1$ and $\mathcal{R}_2$, each divided into sets of tokens $\{X_1, X_2\}$ and $\{Y_1, Y_2\}$, respectively. Let there be a matching set of token pairs $\mathcal{M}$, such that $\mathcal{M} = \{\{X_1, Y_1\}, \{X_2, Y_2\}\}$. For the entire record, the Q-gram similarity threshold is calculated as*

$$t = \max\{|X_1 \cup X_2|, |Y_1 \cup Y_2|\} - q + 1 - qd(X_1 \cup X_2, Y_1 \cup Y_2),$$

which diverges from the separate calculations for each matching pair,

$$t_{\mathcal{M}} = \max\{|X_1|, |Y_1|\} - q + 1 - qd(X_1, Y_1) + \max\{|X_2|, |Y_2|\} - q + 1 - qd(X_2, Y_2),$$

as determined by the edges in a maximum weighted bipartite matching. This demonstrates that $t \neq t_{\mathcal{M}}$, indicating the need for a more refined filter approach for $t_{\mathcal{M}}$.

In [122], a lower bound relationship is established between edit distance and the Q-gram method for a pattern string X of length $|X|$ and a text string Y of length $|Y|$. This lower bound is crucial for string similarity search and similarity join algorithms, which are widely applied in data cleaning, search engines, and data integration [200]. These algorithms, going beyond traditional exact search methods, handle data errors and inconsistencies. Their importance lies in speeding up similarity joins and minimizing exhaustive pairwise comparisons.

The Q-gram Count filter model is refined by introducing more precise assumptions for the token sets $\mathcal{R}_1, \mathcal{R}_2$, and an optimal filter is derived to ensure that no comparison pair with a Fuzzy Jaccard Similarity higher than α is lost.

It is assumed that the matching token pairs provided by $\mathcal{M}$ are known, while their edit distances are unknown. However, these distances can be predicted based on expected edit distances.

Theorem 49 (Optimal Count Q-gram Filter for Bipartite Matching)**.** *Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be records representing a set of tokens. Then the Q-gram similarity in bipartite matching of $\mathcal{R}_1$, $\mathcal{R}_2$ and cardinality $|\mathcal{M}|$ for a given threshold $s_n(\mathcal{R}_1, \mathcal{R}_2) \geq \alpha$ is at least*

$$\begin{aligned} t_{\mathcal{M}} &= \inf_{\alpha}\{|Q_{\mathcal{R}_1} \cap Q_{\mathcal{R}_2}|\} \\ &= \underbrace{\sum_{(i,j)\in\mathcal{M}} \max\{|X_i|, |Y_j|\} - |\mathcal{M}|q + |\mathcal{M}|}_{\textit{maximum shared Q-grams}} \underbrace{-q \max_{\alpha} \sum_{(i,j)\in\mathcal{M}} d(\alpha_{i,j}, |X_i|, |Y_j|)}_{\textit{loss function}}, \end{aligned} \tag{5.68}$$

containing a linear combination of

$$d(\alpha_{i,j}, |X_i|, |Y_j|) = \frac{1 - \alpha_{i,j}}{1 + \alpha_{i,j}}(|X_i| + |Y_j|) \tag{5.69}$$

under the constraint $\alpha = \frac{\sum_{(i,j)\in\mathcal{M}} \alpha_{i,j}}{|\mathcal{R}_1|+|\mathcal{R}_2|-\sum_{(i,j)\in\mathcal{M}} \alpha_{i,j}}$ *for which the linear combination is maximized.*

Proof. Consider the sum over connected pairs of tokens with cardinality $|\mathcal{M}|$

$$\begin{aligned}
&\inf_{\alpha}\{|Q_{\mathcal{R}_1} \cap Q_{\mathcal{R}_2}|\} = \inf_{\alpha}\left\{\sum_{(i,j)\in\mathcal{M}} |Q_{X_i} \cap Q_{Y_i}|\right\} = \sum_{(i,j)\in\mathcal{M}} \inf_{\alpha_{i,j}}\{|Q_{X_i} \cap Q_{Y_j}|\} \\
&= \sum_{(i,j)\in\mathcal{M}} \inf_{\alpha_{i,j}}\{\max\{|X_i|,|Y_j|\} - q + 1 - qd(X,Y)\} \\
&= \sum_{(i,j)\in\mathcal{M}} \{\max\{|X_i|,|Y_j|\} - q + 1 - q\sup_{\alpha_{i,j}} d(X,Y)\} \\
&= \sum_{(i,j)\in\mathcal{M}} \max\{|X_i|,|Y_j|\} - |\mathcal{M}|q + |\mathcal{M}| - q\max_{\alpha} \sum_{(i,j)\in\mathcal{M}} d(\alpha_{i,j},|X_i|,|Y_j|).
\end{aligned} \tag{5.70}$$

Each $\alpha_{i,j}$ is the distributed minimum similarity for each token, giving a threshold vector that should maximize the sum of the expected distances $d(\alpha_{i,j},|X_i|,|Y_j|)$ so that $s_n(\mathcal{R}_1,\mathcal{R}_2) \geq \alpha$ holds for t_M. Formalizing this, the task is obtained

$$\begin{aligned}
&\text{maximize} \quad \sum_{(i,j)\in\mathcal{M}} d(\alpha_{i,j},|X_i|,|Y_j|), \\
&\text{subject to} \quad \sum_{(i,j)\in\mathcal{M}} \alpha_{i,j} \geq \frac{\alpha}{1+\alpha}(|\mathcal{X}|+|\mathcal{Y}|) \quad \alpha\in[0,1], \quad i=1,\ldots,|\mathcal{M}|, \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \alpha_{i,j}\in[0,1], \quad j=1,\ldots,|\mathcal{M}|.
\end{aligned}$$

This leads to an integer linear programming task equivalent to the Knapsack problem, solvable in $\mathcal{O}(nb)$ time. The optimization algorithm determines the maximum expected edit distance distribution across tokens, maintaining the similarity threshold $s_n(\mathcal{R}_1,\mathcal{R}_2) \geq \alpha$. □

The aim is to merge the objective function and its constraint into a single expression using the Lagrange multiplier method

$$\begin{aligned}
&\mathcal{L}(\alpha_{i,j},...,\alpha_{|\mathcal{M}|},\lambda) \\
&= \sum_{(i,j)\in\mathcal{M}} d(\alpha_{i,j},|X_i|,|Y_j|) - \lambda\left(\frac{\alpha}{1+\alpha}(|\mathcal{X}|+|\mathcal{Y}|) - \sum_{(i,j)\in\mathcal{M}} \alpha_{i,j}\right) \\
&= \sum_{(i,j)\in\mathcal{M}} \frac{1-\alpha_{i,j}}{1+\alpha_{i,j}}(|X_i|+|Y_j|) - \lambda\left(\frac{\alpha}{1+\alpha}(|\mathcal{X}|+|\mathcal{Y}|) - \sum_{(i,j)\in\mathcal{M}} \alpha_{i,j}\right).
\end{aligned}$$

and solve $\nabla_{\alpha_{i,j},...,\alpha_{|\mathcal{M}|},\lambda}\mathcal{L}(\alpha_{i,j},...,\alpha_{|\mathcal{M}|},\lambda) = 0$. Differentiating with respect to a specific $\alpha_{i,j}$

and setting the derivative to zero:

$$\frac{\partial \mathcal{L}}{\partial \alpha_{i,j}} = -\frac{2(|X_i| + |Y_j|)}{\alpha_{i,j}^2 + 2\alpha_{i,j} + 1} - \lambda = 0. \tag{5.71}$$

Differentiating with respect to λ and setting this derivative to zero:

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{\alpha}{1+\alpha}(|\mathcal{R}_1| + |\mathcal{R}_2|) - \sum_{(i,j)\in\mathcal{M}} \alpha_{i,j} = 0. \tag{5.72}$$

Solving these equations will yield the values of $\alpha_{i,j}$ and the optimal λ for the given optimization problem. From this, λ can be solved for as follows

$$\lambda = -\frac{2(|X_i| + |Y_j|)}{\alpha_{i,j}^2 + 2\alpha_{i,j} + 1}. \tag{5.73}$$

So the threshold from Theorem (49) can be expressed with the Lagrangian function as follows:

$$\begin{aligned} t_{\mathcal{M}} &= \inf_{\alpha}\{|Q_{\mathcal{R}_1} \cap Q_{\mathcal{R}_2}|\} \\ &= \underbrace{\sum_{(i,j)\in\mathcal{M}} \max\{|X_i|, |Y_j|\} - |\mathcal{M}|q + |\mathcal{M}|}_{\text{maximum shared Q-grams}} - q \underbrace{\mathcal{L}(\alpha_{i,j}, \ldots, \alpha_{|\mathcal{M}|}, \lambda)}_{\text{loss function with Lagrangian}}. \end{aligned} \tag{5.74}$$

5.5.3 Unsupervised Learnable Count Q-gram Filter

Considering the analytical intractability of the optimal count Q-gram filter, the goal is to establish a suitable approximation that maintains accuracy while ensuring computational efficiency, achieving a constant time complexity of $\mathcal{O}(1)$.

The approach is predicated on the following assumptions:

- The cardinalities of the sets $\mathcal{R}_1$, $\mathcal{R}_2$, and the matching set $\mathcal{M}$ are equivalent, i.e., $|\mathcal{R}_1| = |\mathcal{R}_2| = |\mathcal{M}|$.
- For every token pair $(i, j) \in \mathcal{M}$, the length of Y_j is assumed to be the expected value of the lengths of tokens in $\mathcal{R}_2$, represented as $|Y_j| = \mathbb{E}[Y_j]$.
- The similarity threshold α is uniformly applied across all token pairs, such that $\mathbb{E}[\alpha_{i,j}] = \alpha$, and hence $\mathbb{E}[\mathcal{L}(\alpha_{i,j}, ..., \alpha_{|\mathcal{M}|}, \lambda)] = \mathcal{L}(\alpha, \lambda)]$.
- The expected number of destroyed Q-grams, $\mathbb{E}[q_{i,j}]$, is estimated based on the assumption that the edit distance $d = 1$ is a uniformly distributed random variable

over the token. This estimation is represented by the formula:

$$\mathbb{E}[q_{i,j}] = q \cdot \frac{\sup |Q_{X_i} \cap Q_{Y_j}|}{\max\{|X_i|, \mathbb{E}[|Y_j|]\}} = q \cdot \frac{\max\{|X_i|, \mathbb{E}[|Y_j|]\} - q + 1}{\max\{|X_i|, \mathbb{E}[|Y_j|]\}} < q. \tag{5.75}$$

Specifically, by establishing that $\mathbb{E}[q_{i,j}] < q$, the filter criteria become more stringent and so enhancing the selectivity of the filter.

- The mean value of the expected number of destroyed Q-grams across all tokens $\mathcal{R}_1$ is denoted $\mathbb{E}[\mathbb{E}[q_{i,j}]]$.
- The filter sensitivity factor γ is introduced within the range $[0, 1]$, due to certain simplifications and estimations made across the collection of records. This factor is empirically set to be slightly weaker, allowing a balance between high efficiency and precision of the filter.

Under these considerations, the approximation for the Q-gram similarity is derived as follows:

$$\begin{aligned}\mathbb{E}[t_{\mathcal{M}}] &= \sum_{i \in \mathcal{R}_1} \max\{|X_i|, \mathbb{E}[Y_j]\} - |\mathcal{R}_1|q + |\mathcal{R}_1| - \mathcal{L}(\alpha, \lambda) \\ &= \sum_{i \in \mathcal{R}_1} \max\{|X_i|, \mathbb{E}[Y_j]\} - |\mathcal{R}_1|q + |\mathcal{R}_1| - \sum_{i \in \mathcal{R}_1} \mathbb{E}[q_{i,j}] d(\alpha, |X_i|, \mathbb{E}[Y_j]) \\ &\quad - \mathbb{E}[\mathbb{E}[q_{i,j}]] \cdot \gamma \cdot \lambda \cdot \left(\frac{\alpha(1-\alpha)}{1+\alpha} |\mathcal{X}| \right).\end{aligned} \tag{5.76}$$

The loss function $d(\alpha, |X_i|, |Y_j|)$, considering the expected value of Y_j, is defined as:

$$d(\alpha, |X_i|, \mathbb{E}[Y_j]) = \frac{1-\alpha}{1+\alpha}(|X_i| + \mathbb{E}[Y_j]) \tag{5.77}$$

and Lagrange multiplier

$$\lambda = -\frac{2(\mathbb{E}[|X_i|] + \mathbb{E}[Y_j])}{\alpha^2 + 2\alpha + 1}. \tag{5.78}$$

This approximation greatly simplifies the original problem. By standardizing the length of tokens in $\mathcal{R}_2$ to their expected value and using a consistent similarity threshold, the Q-gram similarity calculation is optimized to $\mathcal{O}(1)$. This method is especially useful when the exact lengths of $|Y_j|$ are unknown or when computational efficiency is a priority. The method is implemented in Algorithm 7.

5.5.4 Approximate Count Q-gram Filter

Theorem 50 (Approximate Count Q-gram Filter)**.** *Let $F_{\mathcal{R}_1}$ and $F_{\mathcal{R}_2}$ be discrete distribution functions representing the ascending sorted lengths $|X_i|$ and $|Y_i|$, respectively.*

Consider $\mathcal{R}_1$ and $\mathcal{R}_2$ as sets of records with unknown connected edges and let $\mathcal{M}$ denote the set of matched record pairs with cardinality $|\mathcal{M}|$. Assuming the constancy of $\alpha_{i,j} = \alpha$ for all (i, j) as an approximation, the Q-gram similarity in bipartite matching is at least:

$$\widehat{t}_{\mathcal{M}} = \frac{2q\alpha + \alpha - 2q + 1}{2 + \alpha} \left(F_{\mathcal{R}_1}[|\mathcal{M}|] + F_{\mathcal{R}_2}[|\mathcal{M}|]\right) + \frac{1}{2} \left|F_{\mathcal{R}_1}[|\mathcal{M}|] - F_{\mathcal{R}_2}[|\mathcal{M}|]\right| - |\mathcal{M}|q + |\mathcal{M}| \tag{5.79}$$

for a classification threshold in Fuzzy Bipartite Matching, $s_n(\mathcal{R}_1, \mathcal{R}_2) \geq \alpha$.

Proof. The derivation results from a new approximation method that involves using several techniques to establish a less tight lower bound for certain terms. To facilitate the reader's comprehension, the following equations, integral to the final derived formula, are presented first

$$\max\left\{ \sum_{(i,j)\in\mathcal{M}} |X_i|, \sum_{(i,j)\in\mathcal{M}} |Y_j| \right\} \leq \sum_{(i,j)\in\mathcal{M}} \max\{|X_i|, |Y_j|\}. \tag{5.80}$$

The maximum of any two variables $a, b \in \mathbb{R}$ can also be expressed in another analytical form:

$$\max\{a, b\} = \frac{1}{2}(a + b + |a - b|), \tag{5.81}$$

and now define the cumulative sum (discrete distribution function) of ascending sorted length $F_{\mathcal{R}_1}$ and $F_{\mathcal{R}_2}$. Finally, the inequality is obtained as follows:

$$\max\{F_{\mathcal{R}_1}[|\mathcal{M}|], F_{\mathcal{R}_2}[|\mathcal{M}|]\} \leq \max\left\{ \sum_{(i,j)\in\mathcal{M}} |X_i|, \sum_{(i,j)\in\mathcal{M}} |Y_j| \right\}. \tag{5.82}$$

With equations (5.80), (5.81), and (5.82), the full derivation is proceeded, assuming the constancy of $\alpha_{i,j} = \alpha$ for simplicity. The floor function $\lfloor . \rfloor$ is also applied to maintain integer values for shared Q-grams. The entire derivation is as follows:

$$
\begin{aligned}
t_{\mathcal{M}} &= \sum_{(i,j)\in\mathcal{M}} \max\{|X_i|,|Y_j|\} - |\mathcal{M}|q + |\mathcal{M}| - q\max_{\alpha} \sum_{(i,j)\in\mathcal{M}} d(\alpha,|X_i|,|Y_j|) \\
&= \sum_{(i,j)\in\mathcal{M}} \max\{|X_i|,|Y_j|\} - |\mathcal{M}|q + |\mathcal{M}| - q\max_{\alpha} \sum_{(i,j)\in\mathcal{M}} \frac{1-\alpha_{i,j}}{1+\alpha_{i,j}}(|X_i|+|Y_j|) \\
&\approx \sum_{(i,j)\in\mathcal{M}} \max\{|X_i|,|Y_j|\} - |\mathcal{M}|q + |\mathcal{M}| - q\frac{1-\alpha}{1+\alpha} \sum_{(i,j)\in\mathcal{M}} (|X_i|+|Y_j|) \\
&\geq \max\left\{ \sum_{(i,j)\in\mathcal{M}} |X_i|, \sum_{(i,j)\in\mathcal{M}} |Y_j| \right\} - |\mathcal{M}|q + |\mathcal{M}| - \frac{q-q\alpha}{1+\alpha} \sum_{(i,j)\in\mathcal{M}} (|X_i|+|Y_j|) \\
&= \frac{1}{2}\sum_{(i,j)\in\mathcal{M}} (|X_i|+|Y_j|) + \frac{1}{2}\left| \sum_{(i,j)\in\mathcal{M}} |X_i| - \sum_{(i,j)\in\mathcal{M}} |Y_k| \right| - |\mathcal{M}|q + |\mathcal{M}| - \frac{q-q\alpha}{1+\alpha} \sum_{(i,j)\in\mathcal{M}} (|X_i|+|Y_j|) \\
&= \frac{2q\alpha+\alpha-2q+1}{2+\alpha} \sum_{(i,j)\in\mathcal{M}} (|X_i|+|Y_j|) + \frac{1}{2}\left| \sum_{(i,j)\in\mathcal{M}} |X_i| - \sum_{(i,j)\in\mathcal{M}} |Y_k| \right| - |\mathcal{M}|q + |\mathcal{M}| \\
&\geq \frac{2q\alpha+\alpha-2q+1}{2+\alpha}(F_{\mathcal{R}_1}[|\mathcal{M}|] + F_{\mathcal{R}_2}[|\mathcal{M}|]) + \frac{1}{2}\Big|F_{\mathcal{R}_1}[|\mathcal{M}|] - F_{\mathcal{R}_2}[|\mathcal{M}|]\Big| - |\mathcal{M}|q + |\mathcal{M}| \\
&\geq \left\lfloor \frac{2q\alpha+\alpha-2q+1}{2+\alpha}(F_{\mathcal{R}_1}[|\mathcal{M}|] + F_{\mathcal{R}_2}[|\mathcal{M}|]) + \frac{1}{2}\Big|F_{\mathcal{R}_1}[|\mathcal{M}|] - F_{\mathcal{R}_2}[|\mathcal{M}|]\Big| \right\rfloor - |\mathcal{M}|q + |\mathcal{M}| \\
&= \widehat{t}_{\mathcal{M}} \\
&\implies t_{\mathcal{M}} \approx \widehat{t}_{\mathcal{M}}.
\end{aligned}
\tag{5.83}
$$

□

Consequently, when utilizing a pre-built inverted Q-gram index that includes the distribution of ascending sorted lengths, a time complexity of $\mathcal{O}(1)$ can be achieved.

Theorem 51 (Q-gram Filter Efficiency)**.** *Let α_{optim} be a minimal efficient threshold for a Q-gram filter to be effective at filtering dissimilar records for bipartite matching $s_n(\mathcal{R}_1,\mathcal{R}_2) \geq \alpha$. Then*

$$\alpha_{optim} \geq \frac{2q-1}{2q+1}. \tag{5.84}$$

From Equation (5.82) it follows that $F_X[|\mathcal{M}|] \leq \sum_{i=1}^{|\mathcal{M}|} |X_i|$. It should be proven that in the ordered statistics $|X_1| \leq |X_2|,\ldots,|X_{m-1}| \leq |X_m|$, it is also shown that $t_{X_1} \leq t_{X_2},\ldots,t_{X_{m-1}} \leq t_{X_m}$. For any $|X_i|$ and $|X_j| = |X_i|+1$, the inequality $t_{X_i} \leq t_{X_j}$ holds. Thus, the inequality

$$
\begin{aligned}
|X| - q + 1 - qd(\alpha, |X|, |X|) &\leq (|X| + 1) - q + 1 - qd(\alpha, |X| + 1, |X| + 1), \\
|X| - q + 1 - q\frac{2|X| - 2\alpha|X|}{1+\alpha} &\leq (|X| + 1) - q + 1 - q\frac{2(|X| + 1) - 2\alpha(|X| + 1)}{1+\alpha}, \\
-q\frac{2|X| - 2\alpha|X|}{1+\alpha} &\leq 1 - q\frac{2(|X| + 1) - 2\alpha(|X| + 1)}{1+\alpha}, \\
2|X| - 2\alpha|X| &\geq 2(|X| + 1) - 2\alpha(|X| + 1) - \frac{1+\alpha}{q} \\
\alpha &\geq \frac{2q-1}{2q+1}.
\end{aligned}
\tag{5.85}
$$

As derived previously in (13), the Q-gram filter works for $\alpha \geq \alpha_{sing}$. If $\alpha_{\text{optim}} = \frac{2q-1}{2q+1}$ is put, then the convergence of this is obvious:

$$\lim_{|X|\to\infty} \alpha_{sing}(|X|) = \alpha_{optim}. \tag{5.86}$$

By evaluating the expression, an effective Q-gram filter is obtained for bigram $\alpha \geq 0.6$, for trigram $\alpha \geq \frac{5}{7}$, and for fourgram $\alpha \geq \frac{7}{9}$. The conditions for the minimum thresholds make sense for ordinary use in real applications and have great robustness to errors.

Now, in the previous model, a padding extension is also incorporated, which should further improve the efficiency of the filtering.

Corollary 16 (Optimal Count Q-gram Filter with Padding for Bipartite Matching)**.** *Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be records representing a set of tokens and suppose there is an extension of the tokens by padding them with characters $p \in \Sigma^*$. Then the Q-gram similarity in bipartite matching of $\mathcal{R}_1$, $\mathcal{R}_2$ and cardinality $|\mathcal{M}|$ for given threshold α is at least*

$$t_{\mathcal{M}_p} = \inf_{\alpha,p}\{|Q_{\mathcal{R}_1} \cap Q_{\mathcal{R}_2}|\} = t_{\mathcal{M}} + |\mathcal{M}||p|. \tag{5.87}$$

Proof. Apply Corollary 15 and a procedure similar to that for proving Theorem 49. □

Corollary 17 (Approximate Lower Bound to Optimal Count Q-gram Filter with Padding for Bipartite Matching)**.** *Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be records representing a set of tokens and suppose the number of total padding characters is $|p_{\mathcal{R}_1}|, |p_{\mathcal{R}_2}|$ in the records $\mathcal{R}_1$ and $\mathcal{R}_2$. Then the Q-gram similarity in bipartite matching of $\mathcal{R}_1$, $\mathcal{R}_2$ and cardinality N for given threshold α is at least*

$$t_{\mathcal{M}_p} \approx \widehat{t}_{M_p} = \widehat{t}_{\mathcal{M}} + \min\{|p_{\mathcal{R}_1}|, |p_{\mathcal{R}_2}|\}. \tag{5.88}$$

Proof. Similar to the previous: Use Corollary 15 and Theorem 50. The counting of the total padding characters per each record is due to the existence of tokens shorter than q,

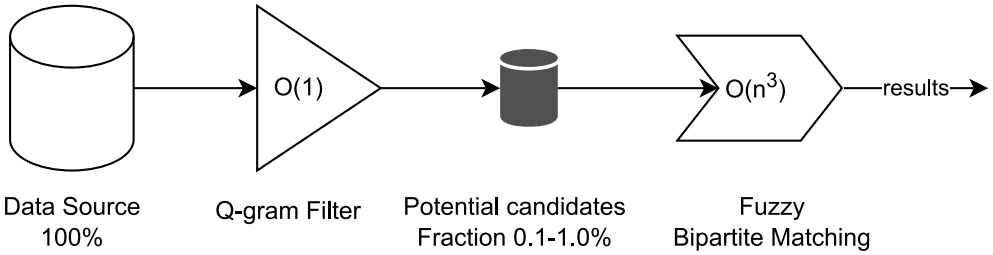

Figure 5.12: Block diagram of the processing of records from the source in real-time by a two-step system of Q-gram filter and Fuzzy Bipartite Matching

where no padding characters are appended, but only the string is prolonged to be at least of size q, ensuring extraction of at least one Q-gram feature. □

5.6 Real-Time Matching and Search in Similarity Space

5.6.1 Software Architecture

The architecture of the application for solving the approximate string-matching problem is shown in Fig. 5.11. The region inside the dashed rectangle delineates the main topics of the thesis. The mathematically derived lower bound of FRS $\widehat{t}_{\mathcal{M}}$ filters dissimilar records $|Q_{\mathcal{R}_1} \cap Q_{\mathcal{R}_2}| < \widehat{t}_{\mathcal{M}}$ from a Q-gram inverted index. The generated candidates only include a very small fraction of the indexed collection of records. This stage mainly guarantees real-time running. Furthermore, FRS runs comparisons on the generated candidates and generates a subset $\mathcal{M}$ of the candidates for $s_n(\mathcal{R}_1, \mathcal{R}_2) \geq \alpha$. This process is illustrated in the block of Fig. 5.12. The critical point is reiterated that a lower threshold α results in a larger set of candidates. The previous chapters explained how the lower bound $\widehat{t}_{\mathcal{M}}$ and FRS are related mathematically to the threshold α [OR-6].

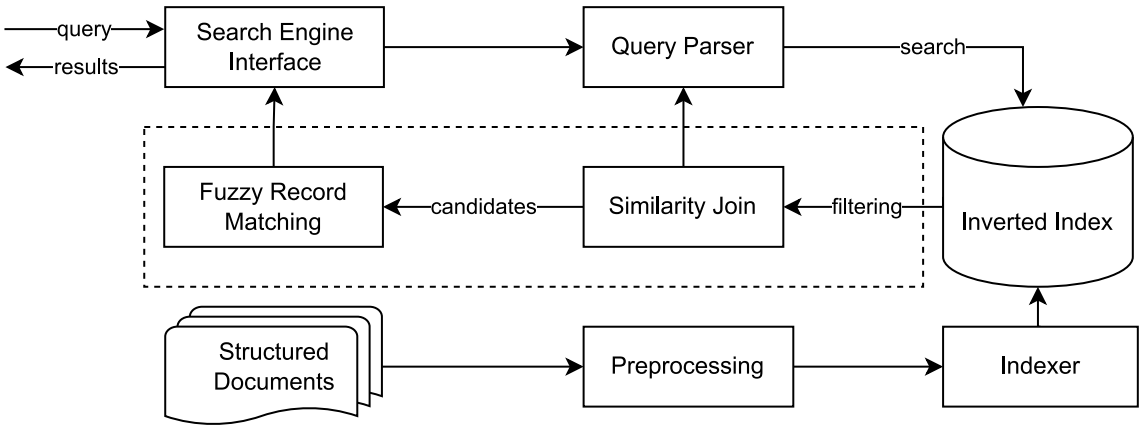

Figure 5.11: Production System Architecture of Fuzzy Search/Matching Engine

5.6.2 Experiments

In this section, the results of an extensive set of experiments conducted to demonstrate the efficiency of the proposed mathematical models for Q-gram filtering and approximate record-weighted matching in a bipartite graph are presented.

The quality of the measurement data can be checked by finding matches in a database created from other sources. A test suite that has been reported in various papers and widely used to analyze these metrics [130], [178] was utilized.

In the dataset, pairs of inputs belonging to the same domain were compared, and if their IDs matched, they were marked as identical. The test involved comparing datasets taken from [130] as shown in Table. 5.3.4. Each dataset was divided into two parts, and between these two parts, scores between all pairs of records were calculated. Subsequently, all pairs were sorted according to the calculated similarity scores. Ideally, all matches should have a higher similarity score and, as a result, should appear in the sorted list before all mismatches.

The *non-interpolated average precision* of this ranking was computed. According to the papers [130], [178], precision and recall were calculated as follows:

$$\text{Precision} = \frac{c(i)}{i}, \tag{5.89}$$

$$\text{Recall} = \frac{c(i)}{m}, \tag{5.90}$$

where $c(i)$ is the number of correct matching pairs ranked before position i, and m is total number of correct matches. Consequently *interpolated precision* at recall r is the $\max_i \frac{c(i)}{i}$, where the max is taken over all ranks i such that $\frac{c(i)}{m} \geq r$. The graphs in Fig. 5.13 and Fig. 5.14 are plotted from the interpolated precision in the recall sequence $r = 0.0$, 0.05, ..., 0.95, 1.0 (21 equidistant recall levels) with a step length of 0.05. The curves go through the points and are smoothed for better clarity. The overall relative performance of the compared similarity functions is calculated using the maximum F1-score as:

$$\text{F1-score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{\text{Precision+Recall}} \times 100\%, \tag{5.91}$$

and shown in the Tab. 5.7. The table shows that the best results of 85.09% were achieved using the FRS method and its combination with the Q-gram filter 85.01%. Based on the results, it can be concluded that such a result confirms the high accuracy of the approximated optimal Q-gram filter and verifies the correctness of the mathematical derivation of its approximate form. However, it should be noted that although it is still an approximation, there are several records for which it was found, upon detailed analysis, that they

did not pass through the filter [OR-6].

A complete ablation study was performed for combinations of the derived Q-gram filters with padding (Theorem 50 and Corollary 17) and the naive Q-gram filter with padding (Corollary 15), along with FRS. Padding $|p| = 2$ indicates that a single prefix and postfix character was used, $|p| = 1$ indicates only a prefix character was used, and $|p| = 0$ no padding was applied. The experiments evaluated whether each compared pair of matches passes the filter at a fixed threshold α corresponding to the final similarity score of the FRS method, i.e. filter threshold $\alpha = s_n(\mathcal{R}_1, \mathcal{R}_2)$. The thresholding parameter α was only reduced by 0.01 as an error tolerance. The effectiveness of the Q-gram filter as a lower bound on the FRS method is tested in this manner [OR-6].

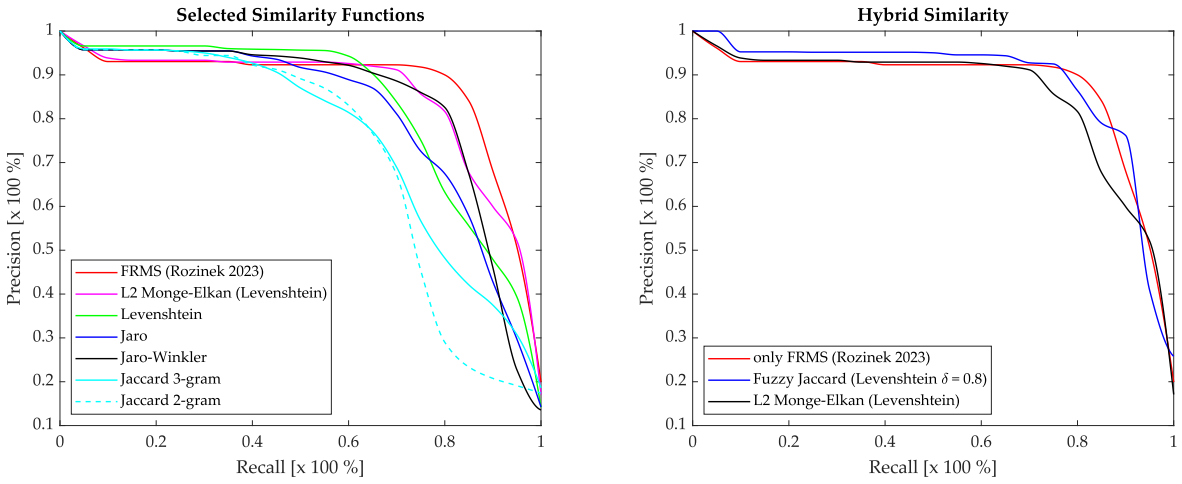

Figure 5.13: Relative performance of selected similarity functions from the group of hybrid, edit and Q-gram similarity

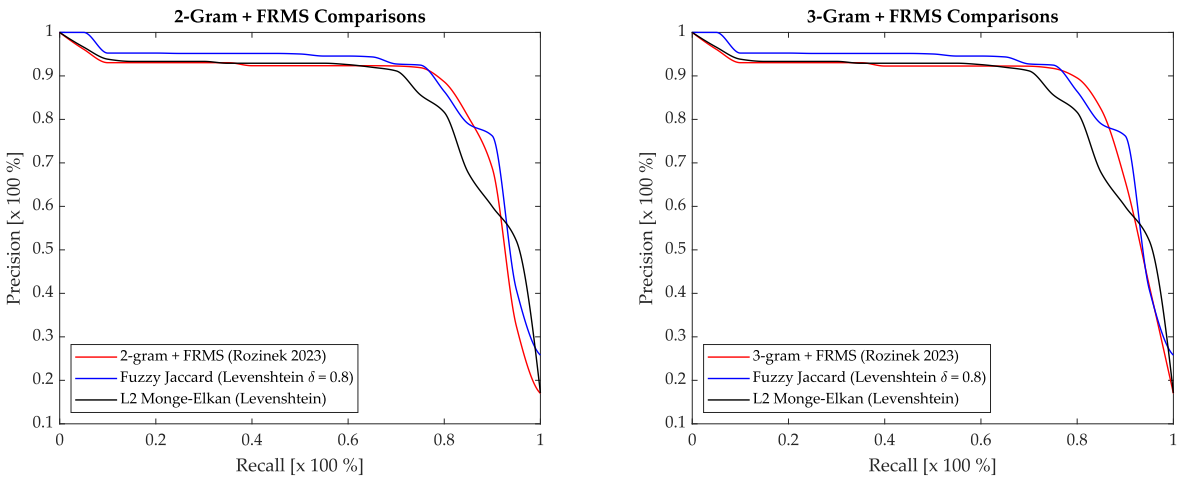

Figure 5.14: Comparison of the relative performance of Q-gram filter+FRS and hybrid similarity functions

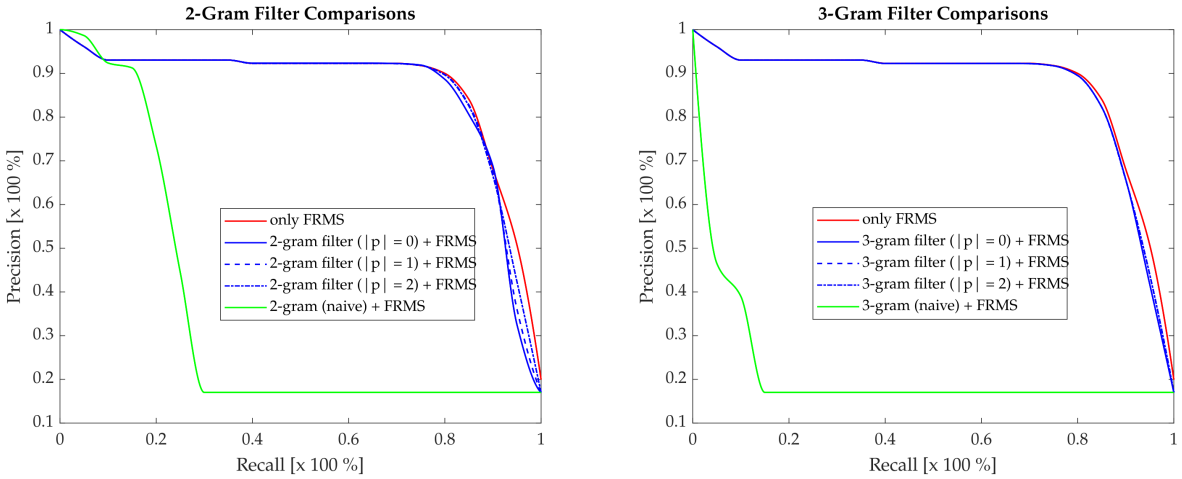

Figure 5.15: Comparison of the relative performance of Q-gram filter+FRS in an ablation study

Similarity	**F1-score**	**Similarity**	**F1-score**
FRS	85.09%	Smith-Waterman	75.71%
3-gram filter+FRS	85.01%	Smith-Waterman-Gotoh	75.54%
2-gram filter+FRS	84.88%	Jaro	75.29%
Fuzzy Jaccard (Levenshtein $\delta = 0.8$))	84.17%	Overlap 3-gram	73.21%
Jaro-Winkler	81.45%	Jaccard 2-gram	71.05%
L2 Monge-Elkan (Levenshtein)	80.80%	Dice 2-gram	71.05%
Damerau-Levenshtein	76.86%	Jaccard 3-gram	70.86%
Levenshtein	76.83%	Dice 3-gram	70.86%
Needleman-Wunsch	76.25%	Overlap 2-gram	66.92%

Table 5.7: F1-score Comparison of selected similarity functions ranked in descending order of F1-score

Similarity	F1-score	Similarity	F1-score
FRS	85.09 %	naive 2-gram ($\|p\| = 2$)+FRS	34.75 %
3-gram ($\|p\| = 2$)+FRS	85.01 %	naive 2-gram ($\|p\| = 1$)+FRS	34.33 %
2-gram ($\|p\| = 2$)+FRS	84.88 %	naive 2-gram ($\|p\| = 0$)+FRS	31.69 %
3-gram ($\|p\| = 1$)+FRS	84.84 %	naive 3-gram ($\|p\| = 2$)+FRS	29.07 %
2-gram ($\|p\| = 1$)+FRS	84.74 %	naive 3-gram ($\|p\| = 1$)+FRS	29.07 %
3-gram ($\|p\| = 0$)+FRS	84.71 %	naive 3-gram ($\|p\| = 1$)+FRS	29.07 %
2-gram ($\|p\| = 0$)+FRS	84.38 %		

Table 5.8: Ablation study of the combination of Q-gram filters with FRS evaluated by F1-score

5.6.3 Time and Space Complexity

As already shown in Fig. 5.12, Q-gram filter and FRS together perform a real-time fuzzy matching run. The speed is affected by the alpha threshold parameter, which affects the size of the pool of potential candidates in the second stage of FRS. Testing is performed to compare relative time complexity on a single-core Intel i7 11370H device with a maximum turbo frequency of 4.80 GHz and 16GB of RAM. The results of these tests are presented in Tab. 5.7. The results of the tests are intended to demonstrate the real-time capacity of the proposed system architecture when compared to the trivial configuration of individual similarity functions. The achieved overall result of 220 milliseconds for the Q-gram filter and FRS is measured for all datasets, and the system architecture shown in the Fig.5.11 is used. Furthermore, the analysis of the time complexity of the Q-gram filter and FRS is dealt with separately.

Similarity	Elapsed Time	Similarity	Elapsed Time
Q-gram Filter+FRS	0s:220ms	Dice 3-gram	10s:717ms
Jaro-Winkler	3s:772ms	Jaccard 2-gram	10s:542ms
Jaro	3s:902ms	Overlap 2-gram	11s:549ms
Jaccard 3-gram	9s:829ms	Dice 2-gram	11s:95ms
Overlap 3-gram	10s:251ms	Fuzzy Jaccard ($\delta = 0.8$)	12s:824ms
Levenshtein	13s:426ms	FRS	13s:474ms
L2 Monge-Elkan (Levenshtein)	14s:209ms	Damerau-Levenshtein	22s:824ms
Needleman-Wunsch	28s:170ms	Smith-Waterman	28s:600ms

Table 5.9: Relative Time Complexity, Sorted by Elapsed Time

If two tokens of sizes $|X_i|$ and $|Y_j|$ are given (keeping the previous notation), the nor-

malized edit similarity is computed by a dynamic algorithm with a time complexity of $\mathcal{O}(|X_i||Y_j|)$ and a space complexity of $\Theta(|X_i||Y_j|)$. Since an implementation highly optimized for performance for production systems is used, the cost matrix is transformed into a short vector, achieving a space complexity of only $\Theta(|Y_j|)$ in this implementation. The construction of the adjacency matrix for the complete bipartite graph takes $\mathcal{O}(|\mathcal{R}_1||\mathcal{R}_2|)$ and requires an allocation $\Theta(|\mathcal{R}_1||\mathcal{R}_2|)$. As discussed before, the solution of the assignment problem by the Kuhn–Munkres algorithm is calculated in addition to the adjacency matrix, incurring $\mathcal{O}(|V|^3)$. For simplicity, let generally denote the number of elements by n. The total time complexity is obtained as follows [OR-6]

$$\mathcal{O}(|\mathcal{R}_1||\mathcal{R}_2|)\mathcal{O}(|X_i||Y_j|) + \mathcal{O}((|V|^3) \approx \mathcal{O}(n^2)\mathcal{O}(n^2) + \mathcal{O}(n^3) \approx \mathcal{O}(n^4). \tag{5.92}$$

5.7 Record Deduplication in Similarity Space

At the core lies the notation of entity (or entity profile), which constitute a uniquely identified description of a real-world object in the form of name-value pairs.

Definition 34 (Entity)**.** *Assuming finite sets of attribute names $n \in \mathcal{N}$, attribute values $\mathcal{V}$, and unique identifiers $i \in \mathcal{I}$. An entity e_i is a tuple (i, A_i), where A_i is a set of name value pairs (n, v) with $v \in (\mathcal{V} \cup \mathcal{I})$. A set of entity $\mathcal{E}$ is called entity collection.*

This definition is adapted for a wide range of (semi-)structured representations, e.g. for the most common language-independent data format JSON, database records or web documents.

Definition 35 (Entity Resolution)**.** *Two entity e_i and e_j match, $e_i \equiv e_j$, if they refer to the same real-world entity. Matching entities are also called duplicates. The task of entity resolution is to find all matching entities within an entity collection or across two or more entity collections.*

The term "entity" is also interchangeable with the term "record" used primarily in databases. In this context, the entity refers to a possible subset of the attributes of the record, and sometimes attributes that are not important for the user definition of the entity can be omitted. For practical reasons, an entity is a subset of a record, denoted $e_i \subseteq R_i$, because for a particular deduplication task, not all attributes are relevant to the record comparison or the attributes are superfluous for a classification.

Definition 36 (Deduplication)**.** *Deduplication is a process represented by a function* $\mathcal{D}\colon \mathcal{E} \to \mathcal{C}$*, where* $\mathcal{E}$ *is a collection of entities, and* $\mathcal{C}$ *is a collection of clusters of duplicate entities within* $\mathcal{E}$*. Each cluster in* $\mathcal{C}$ *consists exclusively of entities that are considered equivalent (duplicates) under a specified equivalence relation* $\equiv$*. Formally, the function is defined as:*

$$\mathcal{D}(\mathcal{E}) = \mathcal{C} = \{\{e_i, \ldots, e_j\}\colon e_i, \ldots, e_j \in \mathcal{E}, \forall e_i \equiv e_j, i \neq j\}. \tag{5.93}$$

The definition based on family of sets $\mathcal{C}$ imposes the necessity to have the output as a clusters of entities with same entity resolution and no singleton entity will be returned. This is the main suggested difference from introduced definition [177].

The need for clusters leads to the use of Euclidean space, which provides an axiomatic system with the properties used by most clustering methods. The Euclidean space, or more generally, a metric space that induces the distance metric, is considered. Distance metrics often serve as optimization criteria for many clustering algorithms. Assuming a non-metric space gives rise to unnatural problems, such as the inability to localize points in the space, measure distances between points, convergence problems in algorithms, slower algorithmic iteration due to relaxing the triangle inequality, and inability to acquire the shape of a cluster, among others. Nevertheless, it has been proven that a quite large family of similarity functions is dual to metric space (e.g., Jaccard index, Tanimoto coefficient, edit similarity, Gaussian similarity, and many others [OR-1]). This dual space is termed a similarity space. Without complex mathematical analysis, such a space can be imagined as an 'inverse space' or a 'symmetric mirror space' with respect to the metric space. Therefore, insistence is placed on such a similarity space where duality to metric spaces has been proven [OR-1].

Definition 37 (Self-Join in Similarity Space)**.** *Given an entity collection* $\mathcal{E}$*, a similarity* $s\colon \mathcal{E}^2 \to \mathbb{R}$*, and a similarity threshold* α*, a similarity join identifies all pairs of entity in* $\mathcal{E}$ *that have similarity at least* α

$$\mathcal{E} \bowtie_\alpha \mathcal{E} = \{(e_i, e_j) \in \mathcal{E}^2\colon s(e_i, e_j) \geq \alpha, i \neq j\}. \tag{5.94}$$

Deduplication is characterized by its effectivness and its efficiency. The first refers to how many true positive duplicates are detected, while the second expresses the computational cost for detecting them. Usually in terms of the number of performed comparisons, which is referred to computational time complexity $\mathcal{O}(D(\mathcal{E}))$. The naive, brute-force approach performs all pairwise comparison on entity collection, having a quadratic complexity $\mathcal{O}(D(\mathcal{E})) = \mathcal{O}(\mathcal{E}^2)$. When deduplicating a single structured data source with $|\mathcal{E}|$ number of entities, the maximum number of comparisons is given by the symmetric matrix with a Cartesian product of $|\mathcal{E}| \times |\mathcal{E}|$. Either the lower or upper triangle is utilized

without the diagonal, as comparison of the entity itself is unnecessary. This results in a final time complexity of $\mathcal{O}(D) = (|\mathcal{E}|^2 - |\mathcal{E}|)/2$, as each entity potentially needs to be compared with all other entities.

To avoid exhaustive pairwise comparisons, the similarity join typically involves two steps of *Filtering* and *Matching*.

Definition 38 (Filtering)**.** *Given an entity collection $\mathcal{E}$, a similarity function $s\colon \mathcal{E}^2 \to \mathbb{R}$, and a similarity threshold α, the Filtering function, denoted as $\mathcal{F}_\alpha$, is a mapping from $\mathcal{E}^2$ to subsets of $\mathcal{E}^2$ defined as:*

$$\mathcal{F}_\alpha = \{(e_i, e_j) \in \mathcal{E}^2 \colon s(e_i, e_j) \geq \alpha, i \neq j\}. \tag{5.95}$$

This function returns a subset of $\mathcal{E}^2$ containing candidate pairs where each pair is potentially similar based on the similarity threshold α.

Definition 39 (Matching)**.** *Given the subset of entity pairs produced by the Filtering function, the Matching function, denoted as $\mathcal{M}_\alpha$, is a mapping from this subset to subsets of $\mathcal{E}^2$ defined as:*

$$\mathcal{M}_\alpha = \{(e_i, e_j) \in \mathcal{F}_\alpha \colon s(e_i, e_j) \geq \alpha\}. \tag{5.96}$$

This function refines the subset output from $\mathcal{F}_\alpha$ and retains only those pairs in which the entities satisfy the similarity criterion defined by the threshold α.

Definition 40 (Self-Join in Similarity Space)**.** *Given an entity collection $\mathcal{E}$, a similarity function $s\colon \mathcal{E}^2 \to \mathbb{R}$, and a similarity threshold α, the Self-Join in a Similarity Space, denoted as $\mathcal{E} \bowtie_\alpha \mathcal{E}$, is the composition of the Filtering function $\mathcal{F}_\alpha$ and the Matching function $\mathcal{M}_\alpha$ and is defined as:*

$$\mathcal{E} \bowtie_\alpha \mathcal{E} = \mathcal{M}_\alpha(\mathcal{F}_\alpha). \tag{5.97}$$

This represents the set of entity pairs from $\mathcal{E}^2$ that meet the similarity threshold α.

5.7.1 Experiments

In these experiments, the expected token length is estimated as the average length across all $\mathcal{E}$ records, denoting $\mathcal{R}_2 \in \mathcal{E}$ as $\mathbb{E}[Y_j] = \overline{Y}$ and $\mathcal{R}_1$ as $\mathbb{E}[X_i] = \overline{X}$. The factor γ is empirically set to 0.75.

The precision and recall metrics are based on the concepts of true positives, false positives, and false negatives:

- *True Positives (TP):* Pairs of records that are correctly placed in the same clusters and belong to the same entity.

- *False Positives (FP):* Pairs of records that are incorrectly placed in the same cluster but belong to different entities.

- *False Negatives (FN):* Pairs of records that belong to the same entity but are incorrectly placed in different clusters.

Based on the defined terms, Precision and Recall are calculated using the formulas: Precision $= \frac{TP}{TP+FP}$ and Recall $= \frac{TP}{TP+FN}$. The F-Score, which is the harmonic mean of Precision and Recall, is given by: $F - Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \times 100\%$.

These equations represent the standard approach to calculating the accuracy of a process such as clustering, balancing the trade-off between precision and recall. The maximum F-Score is calculated over all thresholds α in range $[0, 1]$. For the clustering analysis, DBSCAN [241] and All Nearest Neighbors (ANN) [132] are employed, as shown in Table 5.10. The Q-gram filter achieved a precision of 100%, indicating that no true positive (TP) comparison pairs were erroneously removed, while maintaining a recall of 52%. This high efficiency and filtering capability of the Q-gram filter are evident, as filter passes only twice as many candidate pairs compared to the fraction of TP comparison pairs.

Similarity	**DBSCAN Max F-score**	**ANN Max F-score**
2-Gram Filter + Fuzzy Jaccard	85.20%	69.79%
3-Gram Filter + Fuzzy Jaccard	85.20%	69.79%
Fuzzy Jaccard	85.20%	69.79%
2-Gram Jaccard	82.48%	60.61%
Jaro	79.17%	71.63%
2-Gram Overlap	65.86%	79.17%
3-Gram Jaccard	78.86%	66.19%
3-Gram Overlap	67.20%	74.05%
Jaro-Winkler	73.33%	61.69%
Levenshtein	68.59%	55.16%

Table 5.10: Sorted Comparison of Max F-scores for DBSCAN and All Nearest Neighbors Clustering Algorithms on Labelled Vauniv Dataset (116 Records and 15 Clusters) [130]

The main contribution surpasses existing state-of-the-art models with the development of an optimal Q-gram Count filter for bipartite matching, ensuring no true positive (TP) comparison pairs are lost, as the lower bound of bipartite matching is matehmatically derived. Given its analytical intractability, a precise estimation method for this filter is proposed, which operates in constant time complexity $\mathcal{O}(1)$. In these tests, this approach

achieved 100% precision in filtering with a high filtering capability. Altogether, the proposed extended count Q-gram filter significantly speeds up the process of similarity join while maintaining high filter efficiency and precision. The record deduplication was efficiently conducted using DBSCAN clustering, which has the significant advantage of being able to form clusters of arbitrary shape while maintaining fast performance, characterized by a time complexity of $\mathcal{O}(|\mathcal{E}|\log(|\mathcal{E}|))$ [241]. It is worth noting that scalability can be achieved by using one of the parallel versions of DBSCAN, as discussed in [242].

Chapter 6

Discussion

The main contribution of the author in this thesis lies in the further development of the theory of similarity space, demonstrating the class of functions that reside within the similarity space $\mathcal{C}(A)$ (section 2.7). Significant effort has been dedicated to establishing proper transformations between similarity and metric spaces (section 2.5), along with the introduction of the Generalized Rozinek similarity solution to an open, previously unsolved problem (section 2.7). Additionally, the thesis explores the intriguing concept of s-continuity (section 2.4), introduces novel fixed-point theories with practical applications in differential equations (subsection 4.5.1) and the Newton method (subsection 4.5.2), and presents a new approach to linear regression (chapter 3). The author also unveils substantial potential for embedding into measure spaces (subsection 2.6.1), probability spaces (subsection 2.6.2), and Hilbert spaces (subsection 2.6.3), while showcasing promising directions in establishing a new s-norm and enriching functional analysis with novel insights (subsection 2.6.3).

The results presented in the Table 6 underscore the significant contributions of the author to the domain of similarity functions in NLP tasks, as highlighted by the entries denoted in blue. Time complexities are simplified for an illustrative overview of each algorithm.

The author's contributions in approximate string matching, especially with ConvJ and ConvJW (section 5.3), surpass existing state-of-the-art methods in both efficiency and effectiveness. Achieving an F1-score of 87.95% with ConvJW, these methods leverage convolutional approaches and the Fast Fourier Transform (FFT), enabling a theoretical time complexity of $\mathcal{O}(n \log n)$. This approach underlines the potential for handling large datasets effectively. Additionally, fuzzy record matching methods marked as $\mathcal{O}(1)$, indicating constant time due to a prebuilt index, showcase the fastest execution times.

The FRS method, despite its higher time complexity of $\mathcal{O}(n^4)$, still secures an F1-score of 85.09%, illustrating a balance between computational demand and accuracy in similar-

ity assessment. The novel use of "3-gram filter+FRS" and "2-gram filter+FRS" methods further reflects the author's innovative strategy to boost performance, maintaining high F1-scores through effective filtering and similarity calculations.

In contrast, conventional methods like Levenshtein and Jaro-Winkler, while foundational, exhibit lower F1-scores in this analysis, highlighting the advanced nature of the author's techniques. These contributions significantly push forward the field of approximate string matching, offering solutions that are not only technically sophisticated but also highly applicable to real-world data challenges.

Similarity Function	**Time Complexity**	**F1-Score**	**Character-Based**	**Token-Based**
ConvJW($\sigma = 2, \beta = 0.1$)	$\mathcal{O}(n \log n)$	87.95%	✓	
FRS	$\mathcal{O}(n^4)$	85.09%		✓
3-gram filter+FRS	$\mathcal{O}(1)$	85.01%		✓
ConvJ($\sigma = 1$)	$\mathcal{O}(n \log n)$	84.99%	✓	
2-gram filter+FRS	$\mathcal{O}(1)$	84.88%		✓
Fuzzy Jaccard	$\mathcal{O}(n^4)$	84.17%		✓
Jaro-Winkler	$\mathcal{O}(n^2)$	81.45%	✓	
L2 Monge-Elkan	$\mathcal{O}(n^3)$	80.80%		✓
Damerau-Levenshtein	$\mathcal{O}(n^2)$	76.86%	✓	
Levenshtein	$\mathcal{O}(n^2)$	76.83%	✓	
Needleman-Wunsch	$\mathcal{O}(n^2)$	76.25%	✓	
Smith-Waterman	$\mathcal{O}(n^2)$	75.71%	✓	
Smith-Waterman-Gotoh	$\mathcal{O}(n^2)$	75.54%	✓	
Jaro	$\mathcal{O}(n^2)$	75.29%	✓	
Overlap 3-gram	$\mathcal{O}(n)$	73.21%	✓	
Jaccard 2-gram	$\mathcal{O}(n)$	71.05%	✓	
Dice 2-gram	$\mathcal{O}(n)$	71.05%	✓	
Jaccard 3-gram	$\mathcal{O}(n)$	70.86%	✓	
Dice 3-gram	$\mathcal{O}(n)$	70.86%	✓	
Overlap 2-gram	$\mathcal{O}(n)$	66.92%	✓	

Table 6.1: Comparison of similarity functions based on F1-score and their respective time complexities, highlighting both character-based and token-based approaches. The author's contributions are indicated by the text colored in blue.

Chapter 7

Conclusion

This thesis is focused on research of the theory o similarity space and its application, especially in NLP. The central research question and motivation for this thesis is essentially simple: *What is an ideal similarity function and what are its properties?*

The current state-of-the-art is covered of similarity and similarity space and its current definitions and properties in the theoretical part (section 2.1 and section 2.2). The current state-of-the-art shows that this is a new theory that has not yet been fully exploited and does not have clearly stated conventions.

This dissertation successfully addresses the ambitious objectives set forth in section 1.3, advancing the field of similarity space both theoretically and practically. The theoretical contributions, detailed in the respective sections of the dissertation, have significantly expanded our understanding and application of similarity space. Notably, the introduction of a formal definition of similarity space and its axiomatic systems (section 1.3) marks a pivotal advancement, placing similarity space alongside metric spaces in mathematical significance. The exploration of duality between similarity and metric spaces (section 2.5), along with the development of novel theorems such as the linear transformation and convex combination of normalized similarity, underscores the depth of theoretical exploration undertaken.

The practical implications of this work are profound, particularly in the realms of approximate string matching, record matching, and deduplication within similarity space. The development of convolutional-based string matching (chapter 5) represent groundbreaking contributions to the field, offering enhanced accuracy and efficiency over state-of-the-arts methods. Additionally, the creation of the Fuzzy Record Similarity and the Optimal Q-gram Filter further exemplifies the successful application of theoretical concepts to solve real-world problems, reinforcing the interconnectedness of theory and practice in this research.

This dissertation does not merely bridge the gap between theoretical knowledge and

practical application; it also proposes a forward-looking direction for future research in the field. The introduction of s-norm in functional analysis (chapter 2) and the novel application of fixed-point theory in similarity space (chapter 4) open new avenues for exploration, promising to inspire subsequent innovations.

In conclusion, this dissertation accomplishes its stated goals by unifying and further developing the theory of similarity space and demonstrating its broad applicability through practical implementations. The seamless integration of theory and practice not only validates the relevance of mathematical theories to real-world challenges but also sets a precedent for future research at the intersection of these domains. The contributions made herein not only enhance the academic understanding of similarity spaces but also offer tangible tools and methodologies for tackling complex problems in approximate string matching, record matching, and deduplication, among others. As this work aptly demonstrates, the pursuit of theoretical advancement in tandem with practical application remains a cornerstone of meaningful scientific inquiry.

The research was further conducted in cooperation with the software company Rozinet s.r.o. in the form of a case study, where the theoretical conclusions from the initial analysis were verified and implemented in commercial and government projects.

Appendix A

Algorithms

A.1 Simple Linear Regression in Similarity Space

Algorithm 1 Optimization using Dot Product Maximization with Regularization

Input: Data points $(x_1, y_1), \ldots, (x_n, y_n)$, Regularization coefficient λ, Learning rate α, Tolerance ϵ, Maximum iterations $max\_iterations$
Output: Parameters θ_0, θ_1
Initialize θ_0, θ_1 to some starting values
Initialize J_{old} to a large value
$iteration \leftarrow 0$
while $iteration < max\_iterations$ **do**
    $J \leftarrow \sum_{i=1}^{n} y_i(\theta_0 + \theta_1 x_i) - \lambda \sum_{i=1}^{n} (\theta_0 + \theta_1 x_i)^2$
    if $|J - J_{old}| < \epsilon$ **then break**
    $J_{old} \leftarrow J$
    $\nabla J_0 \leftarrow \sum_{i=1}^{n} y_i - 2\lambda(\theta_0 + \theta_1 x_i)$
    $\nabla J_1 \leftarrow \sum_{i=1}^{n} y_i x_i - 2\lambda x_i(\theta_0 + \theta_1 x_i)$
    $\theta_0 \leftarrow \theta_0 + \alpha \nabla J_0$
    $\theta_1 \leftarrow \theta_1 + \alpha \nabla J_1$
    $iteration \leftarrow iteration + 1$
return θ_0, θ_1

A.2 Convolution-Based String Matching

Algorithm 2 Jaro (Implementation Rosetta [228])

Require: $S1, S2$ ▷ Two input strings
Ensure: $s_J(S1, S2)$ ▷ Normalized similarity score between 0 and 1

if $S1 = \text{NULL}$ or $S2 = \text{NULL}$ **then**
    return 0.0
$|S1| \leftarrow$ length of $S1$
$|S2| \leftarrow$ length of $S2$
$w \leftarrow \max(|S1|, |S2|)/2 - 1$
Initialize $s1Matches[1 \dots |S1|]$ to all **false**
Initialize $s2Matches[1 \dots |S2|]$ to all **false**
$m \leftarrow 0$ ▷ Number of matches
$t \leftarrow 0$ ▷ Half the number of transpositions
for $i = 0$ to $|S1| - 1$ **do**
    for $j = \max(0, i - w)$ to $\min(i + w + 1, |S2|) - 1$ **do**
        if $s2Matches[j]$ is true or $S1[i] \neq S2[j]$ **then**
            continue
        $s1Matches[i] \leftarrow$ true
        $s2Matches[j] \leftarrow$ true
        $m \leftarrow m + 1$
        break
if $m = 0$ **then**
    return 0.0
$k \leftarrow 0$
for $i = 0$ to $|S1| - 1$ **do**
    if $s1Matches[i]$ **then**
        while $s2Matches[k]$ is false **do**
            $k \leftarrow k + 1$
        if $S1[i] \neq S2[k]$ **then**
            $t \leftarrow t + 1$
        $k \leftarrow k + 1$
$s_J(S1, S2) \leftarrow \frac{1}{3}\left(\frac{m}{|S1|} + \frac{m}{|S2|} + \frac{m - t/2}{m}\right)$
return $s_J(S1, S2)$

Algorithm 3 Jaro-Winkler (Implementation Rosetta [228])

Require: $S1, S2$ ▷ Two input strings
Ensure: $s_{JW}(S1, S2)$ ▷ Normalized similarity score between 0 and 1

$s_J \leftarrow$ Jaro Similarity$(S1, S2)$
$l \leftarrow 0$
for $i = 0$ to $\min(\min(|S1|, |S2|), 4) - 1$ **do**
    if $S1[i] = S2[i]$ **then**
        $l \leftarrow l + 1$
    else
        break
$p \leftarrow 0.1$ ▷ Scaling factor
$s_{\mathrm{JW}} \leftarrow s_J + l \cdot p \cdot (1 - s_J)$
return s_{JW}

Algorithm 4 Convolutional Jaro (ConvJ)

Require: S_1, S_2 ▷ Two input strings
Require: σ ▷ Standard deviation for Gaussian weighting
Ensure: s_J ▷ Normalized similarity score [0,1]

$w \leftarrow \lceil 3.29 \cdot \sigma \rceil$
$G \leftarrow \text{PrecomputeGaussian}(w, \sigma)$
function PRECOMPUTEGAUSSIAN(w, σ)
    Initialize a 1D array $G[0 \dots w]$
    for $d = 0$ to w **do**
        $G[d] \leftarrow \exp\left(-\frac{d^2}{2\sigma^2}\right)$
    return G
$M_w \leftarrow 0$ ▷ Sum of weights for matches
$A_w \leftarrow 0$ ▷ Sum of weights for misalignments
for $i = 0$ to $|S_1| - 1$ **do**
    $M(i) \leftarrow 0$
    for $j = \max(0, i - w)$ to $\min(|S_2|, i + w + 1) - 1$ **do**
        if $S_1[i] = S_2[j]$ **then**
            $weight \leftarrow G[|i - j|]$
            $M(i) \leftarrow \max(M(i), weight)$
            if $weight = 1.0$ **then**
                break ▷ Early termination if perfect match
    $M_w \leftarrow M_w + M(i)$
    if $M(i) > 0$ and $S_1[i] \neq S_2[j]$ **then**
        $A_w \leftarrow A_w + M(i)$
$s_J \leftarrow \frac{1}{3}\left(\frac{M_w}{|S_1|} + \frac{M_w}{|S_2|} + \frac{M_w - A_w}{M_w}\right)$
return s_J

Algorithm 5 Convolutional Jaro-Winkler (ConvJW)

Require: S_1, S_2 ▷ Two input strings
Require: σ ▷ Standard deviation for Gaussian
Require: β ▷ Decay rate for exponential weighting
Ensure: s_{JW} ▷ Normalized similarity score [0,1]

$w \leftarrow \lceil 3.29 \cdot \sigma \rceil$
$G \leftarrow \text{PrecomputeGaussian}(w, \sigma)$
$E \leftarrow \text{PrecomputeExponential}(\max(|S_1|, |S_2|), \beta)$
$C \leftarrow \text{PrecomputeMaxSum}(\max(|S_1|, |S_2|), \beta)$
function PrecomputeGaussian(w, σ)
    Initialize a 1D array $G[0 \dots w]$
    for $d = 0$ to w **do**
        $G[d] \leftarrow \exp\left(-\frac{d^2}{2\sigma^2}\right)$
    return G
function PrecomputeExponential($maxLength, \beta$)
    Initialize a 1D array $E[0 \dots maxLength]$
    for $i = 0$ to $maxLength$ **do**
        $E[i] \leftarrow \exp(-\beta \cdot i)$
    return E
function PrecomputeMaxSum($maxLength, \beta$)
    Initialize a variable $sum \leftarrow 0$
    Initialize a 1D array $C[0 \dots maxLength]$
    for $i = 0$ to $maxLength$ **do**
        $sum \leftarrow sum + \exp(-\beta \cdot i)$
        $C[i] \leftarrow sum$ ▷ Cumulative sum up to index i
    return $MaxSum$
$M_w \leftarrow 0$ ▷ Sum of weights for matches
$A_w \leftarrow 0$ ▷ Sum of weights for misalignments
for $i = 0$ to $|S_1| - 1$ **do**
    $M(i) \leftarrow 0$
    for $j = \max(0, i - w)$ to $\min(|S_2|, i + w + 1) - 1$ **do**
        if $S_1[i] = S_2[j]$ **then**
            $weight \leftarrow G[|i - j|] \cdot E[\min(i, j)]$
            $M(i) \leftarrow \max(M(i), weight)$
            if $weight = E[\min(i, j)]$ **then**
                break ▷ Early termination if perfect match
    $M_w \leftarrow M_w + M(i)$
    if $M(i) > 0$ and $i \neq j$ **then**
        $A_w \leftarrow A_w + M(i)$
$s_{JW} \leftarrow \frac{1}{3}\left(\frac{M_w}{C[|S_1|]} + \frac{M_w}{C[|S_2|]} + \frac{M_w - A_w}{M_w}\right)$
return s_{JW}

A.3 Real-Time Matching and Search in Similarity Space

Algorithm 6 Fuzzy Overlap Similarity

Input: Records $\mathcal{R}_1$, $\mathcal{R}_2$, Similarity threshold α
Output: Fuzzy Overlap Similarity $sim_O(\mathcal{R}_1, \mathcal{R}_2)$
procedure CalculateFuzzyOverlap($\mathcal{R}_1$, $\mathcal{R}_2$, α)
    Initialize similarity score $sim_O(\mathcal{R}_1, \mathcal{R}_2) \leftarrow 0$
    Create a bipartite graph $\mathcal{G}$ with nodes from $\mathcal{R}_1$ and $\mathcal{R}_2$
    for each pair of tokens (X_i, Y_j) where $X_i \in \mathcal{R}_1$ and $Y_j \in \mathcal{R}_2$ **do**
        Calculate token similarity $s(X_i, Y_j)$
        Add edge (X_i, Y_j) to $\mathcal{G}$ with weight $s_n(X_i, Y_j)$
    Compute maximum weighted matching $\mathcal{M}$ in $\mathcal{G}$ using Kuhn-Munkres algorithm
    $sim_O(\mathcal{R}_1, \mathcal{R}_2) \leftarrow \frac{\sum_{(i,j)\in\mathcal{M}} s_n(X_i, Y_j)}{|\mathcal{M}|}$
    return $sim_O(\mathcal{R}_1, \mathcal{R}_2)$

Algorithm 7 UL-BipartiteJoin: Two-Step Unsupervised Learnable Similarity Join Approach

Input: Query record $\mathcal{R}_1$, Records $\mathcal{R}_2 \in \mathcal{E}$, Inverted Q-gram Index $\mathcal{I}$, Distribution of token lengths $F_{\mathcal{R}_2}$, Q-gram length q, Similarity threshold α
Output: Matched Record Pairs
procedure BipartiteJoin($\mathcal{R}_1$, $\mathcal{I}$, $F_{\mathcal{R}_2}$ q, α)
    $\widehat{t}_{\mathcal{M}} \leftarrow$ CalculateULQGramFilter$(\mathcal{R}_1, \mathcal{I}, F_{\mathcal{R}_2}, q, \alpha)$
    Initialize list of matched pairs $\mathcal{R} \leftarrow []$
    for each record $\mathcal{R}_2$ in $\mathcal{E}, \mathcal{I}$ **do**
        if Q-gram similarity of $|\mathcal{R}_1 \cap \mathcal{R}_2| \geq \widehat{t}_{\mathcal{M}}$ **then**
            $\text{sim}_J(\mathcal{R}_1, \mathcal{R}_2) \leftarrow$ CalculateFuzzyJaccard$(\mathcal{R}_1, \mathcal{R}_2, \alpha)$
            if $\text{sim}_J(\mathcal{R}_1, \mathcal{R}_2) \geq \alpha$ **then**
                Add tuple $(\mathcal{R}_1, \mathcal{R}_2)$ to $\mathcal{R}$
    return $\mathcal{R}$

Algorithm 8 BipartiteJoin: Two-Step Similarity Join Approach

Input: Query record $\mathcal{R}_1$, Records $\mathcal{R}_2 \in \mathcal{E}$, Inverted Q-gram Index $\mathcal{I}$, Distribution of token lengths $F_{\mathcal{R}_2}$, Q-gram length q, Similarity threshold α
Output: Matched Record Pairs
procedure BipartiteJoin($\mathcal{R}_1$, $\mathcal{I}$, $F_{\mathcal{R}_2}$ q, α)
    $\widehat{t}_{\mathcal{M}} \leftarrow$ CalculateAprroxQGramFilter($\mathcal{R}_1, \mathcal{I}, F_{\mathcal{R}_2}, q, \alpha$)
    Initialize list of matched pairs $\mathcal{R} \leftarrow []$
    for each record $\mathcal{R}_2$ in $\mathcal{E}, \mathcal{I}$ **do**
        if Q-gram similarity of $|\mathcal{R}_1 \cap \mathcal{R}_2| \geq \widehat{t}_{\mathcal{M}}$ **then**
            $S \leftarrow$ CalculateFuzzyOverlap($\mathcal{R}_1, \mathcal{R}_2, \alpha$)
            if $S \geq \alpha$ **then**
                Add tuple $(\mathcal{R}_1, \mathcal{R}_2)$ to $\mathcal{R}$
    return $\mathcal{R}$

Algorithm 9 Approximate Count Q-gram Filter

Input: Query record $\mathcal{R}_1$, Inverted Q-gram Index $\mathcal{I}$, Distribution of token lengths $F_{\mathcal{R}_2}$ Q-gram length q, Similarity threshold α
Output: Approximate Count Q-gram Filter Threshold $\widehat{t}_{\mathcal{M}}$
procedure CalculateAprroxQGramFilter($\mathcal{R}_1$, $\mathcal{I}$, $F_{\mathcal{R}_2}$ q, α)
    Initialize $\widehat{t}_{\mathcal{M}} \leftarrow 0$ ▷ Initialize approximate count Q-gram filter
    for each token X_i in $\mathcal{R}_1$ **do**
        Generate Q-grams for X_i
        for each Q-gram Q of X_i **do**
            if Q exists in Inverted Index $\mathcal{I}$ **then**
                Update $\widehat{t}_{\mathcal{M}}$ using the distribution of token lengths in $\mathcal{I}$
    Calculate $F_{\mathcal{R}_1}$ for query $\mathcal{R}_1$ ▷ Cumulative sum of ascending sorted lengths
    $\widehat{t}_{\mathcal{M}} \leftarrow \left\lfloor \frac{2q\alpha+\alpha-2q+1}{2+\alpha}(F_{\mathcal{R}_1} + F_{\mathcal{R}_2}) + \frac{1}{2}\left|F_{\mathcal{R}_1} - F_{\mathcal{R}_2}\right| \right\rfloor - |\mathcal{M}|q + |\mathcal{M}|$
    return $\widehat{t}_{\mathcal{M}}$

Algorithm 10 Unsupervised Learnable Count Q-gram Filter

Input: Records $\mathcal{R}_1$, $\mathcal{R}_2$, Expected length of tokens $\mathbb{E}[|Y_j|]$, Similarity threshold α, Q-gram length q, Filter sensitivity factor γ
Output: Unsupervised Learnable Count Q-gram Filter Threshold $\mathbb{E}[t_{\mathcal{M}}]$
procedure CalculateULQGramFilter($\mathcal{R}_1$, $\mathcal{R}_2$, $\mathbb{E}[|Y_j|]$, α, q, γ)
    $\mathbb{E}[t_{\mathcal{M}}] \leftarrow 0$ ▷ Initialize probabilistic count Q-gram filter
    Calculate $\mathbb{E}[\mathbb{E}[q_{i,j}]]$
    for each token X_i in $\mathcal{R}_1$ **do**
        $\mathbb{E}[q_{i,j}] \leftarrow q \cdot \frac{\max\{|X_i|,\mathbb{E}[|Y_j|]\}-q+1}{\max\{|X_i|,\mathbb{E}[|Y_j|]\}}$
        $\lambda \leftarrow -\frac{2(\mathbb{E}[|X_i|]+\mathbb{E}[|Y_j|])}{\alpha^2+2\alpha+1}$
        Update $\mathbb{E}[t_{\mathcal{M}}] \leftarrow \sum_{i\in\mathcal{R}_1} \max\{|X_i|,\mathbb{E}[|Y_j|]\} - |\mathcal{R}_1|q + |\mathcal{R}_1| - \mathcal{L}(\alpha,\lambda)$
    return $\mathbb{E}[t_{\mathcal{M}}]$

Algorithm 11 Fuzzy Jaccard Similarity

Input: Records $\mathcal{R}_1$, $\mathcal{R}_2$, Similarity threshold α
Output: Fuzzy Jaccard Similarity $\text{sim}_J(\mathcal{R}_1,\mathcal{R}_2)$
procedure CalculateFuzzyJaccard($\mathcal{R}_1$, $\mathcal{R}_2$, α)
    Initialize similarity score $\text{sim}_J(\mathcal{R}_1,\mathcal{R}_2) \leftarrow 0$
    Create a bipartite graph $\mathcal{G}$ with nodes from $\mathcal{R}_1$ and $\mathcal{R}_2$
    for each pair of tokens (X_i, Y_j) where $X_i \in \mathcal{R}_1$ and $Y_j \in \mathcal{R}_2$ **do**
        Calculate token similarity $s(X_i, Y_j)$
        Add edge (X_i, Y_j) to $\mathcal{G}$ with weight $s_n(X_i, Y_j)$
    Compute maximum weighted matching $\mathcal{M}$ in $\mathcal{G}$ using Kuhn-Munkres algorithm
    $\text{sim}_J(\mathcal{R}_1,\mathcal{R}_2) \leftarrow \frac{\sum_{(i,j)\in\mathcal{M}} s_n(X_i,Y_j)}{|\mathcal{R}_2|+|\mathcal{R}_1|-\sum_{(i,j)\in\mathcal{M}} s_n(X_i,Y_j)}$
    return $\text{sim}_J(\mathcal{R}_1,\mathcal{R}_2)$

Algorithm 12 Building Inverted Q-gram Index

Input: Set of records $\mathcal{E}$, Q-gram length q
Output: Inverted Q-gram Index $\mathcal{I}$
procedure BuildInvertedQGramIndex($\mathcal{E}$, q)
    Initialize empty index $\mathcal{I}$
    for each record $\mathcal{R}_2$ in $\mathcal{E}$ **do**
        for each token Y_j in $\mathcal{R}_2$ **do**
            Generate Q-grams of length q from Y_j
            for each Q-gram Q in Q-grams of Y_j **do**
                if Q not in $\mathcal{I}$ **then**
                    Initialize an empty hash table in $\mathcal{I}[Q]$
                if record identifier of $\mathcal{R}_2$ not in $\mathcal{I}[Q]$ **then**
                    Initialize frequency $\mathcal{I}[Q]$[identifier of $\mathcal{R}_2$] $\leftarrow 0$
                Increment frequency count $\mathcal{I}[Q]$[identifier of $\mathcal{R}_2$]
    return $\mathcal{I}$

Algorithm 13 Searching in Inverted Q-gram Index with QGramCount Algorithm

Input: Query record $\mathcal{R}_1$, Inverted Q-gram Index $\mathcal{I}$, Q-gram length q, Similarity threshold τ
Output: Set of matching records $\mathcal{R}$
procedure SEARCHINVERTEDQGRAMINDEX($\mathcal{R}_1$, $\mathcal{I}$, q, τ)
    Initialize an empty count map C ▷ To store Q-gram counts per record
    Initialize an empty set $\mathcal{R}$ ▷ For results
    $Q_{\mathcal{R}_1} \leftarrow$ Generate Q-grams of length q from $\mathcal{R}_1$ ▷ Store Q-grams of $\mathcal{R}_1$
    for each Q-gram Q in $Q_{\mathcal{R}_1}$ **do**
        if Q in $\mathcal{I}$ **then**
            for each record identifier id in $\mathcal{I}[Q]$ **do**
                if id not in C **then**
                    $C[id] \leftarrow 0$ ▷ Initialize count for new id
                $C[id] \leftarrow C[id] + 1$ ▷ Increment count for this id
    for each id, $count$ in C **do**
        if $count \geq \tau$ **then**
            Add id to $\mathcal{R}$ ▷ Add id to results if it meets threshold
    return $\mathcal{R}$ ▷ Return the set of results

A.4 Record Deduplication in Similarity Space

Algorithm 14 Record Deduplication using DBSCAN

Input: Set of records $\mathcal{E}$, Similarity threshold α, MinPts = 1
Output: Set of clusters $\mathcal{C}$, representing deduplicated records
procedure DBSCANDEDUPLICATION($\mathcal{E}$, α)
    Initialize set of clusters $\mathcal{C} \leftarrow \emptyset$
    Mark all records in $\mathcal{E}$ as unvisited
    for each record e_i in $\mathcal{E}$ **do**
        if e_i is unvisited **then**
            Mark e_i as visited
            $\mathcal{N} \leftarrow$ FINDNEIGHBORS(e_i, $\mathcal{E}$, α)
            if size of $\mathcal{N} \geq$ MinPts **then**
                Initialize a new cluster $\mathcal{K}$
                EXPANDCLUSTER(e_i, $\mathcal{N}$, $\mathcal{K}$, α)
                Add $\mathcal{K}$ to $\mathcal{C}$
    return $\mathcal{C}$
procedure FINDNEIGHBORS(e_i, $\mathcal{E}$, α)
    Initialize empty list $\mathcal{N}$
    for each record e_j in $\mathcal{E}$ **do**
        if SIMILARITY(e_i, e_j) $\geq \alpha$ **then**
            Add e_j to $\mathcal{N}$
    return $\mathcal{N}$
procedure EXPANDCLUSTER(e_i, $\mathcal{N}$, $\mathcal{K}$, α)
    Add e_i to cluster $\mathcal{K}$
    for each record e_j in $\mathcal{N}$ **do**
        if e_j is unvisited **then**
            Mark e_j as visited
            $\mathcal{N}' \leftarrow$ FINDNEIGHBORS(e_j, $\mathcal{E}$, α)
            if size of $\mathcal{N}' \geq$ MinPts **then**
                $\mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{N}'$
        if e_j is not yet a member of any cluster **then**
            Add e_j to cluster $\mathcal{K}$

Algorithm 15 Record Deduplication using All Nearest Neighbors (ANN)

Input: Set of records $\mathcal{E}$, Similarity threshold α
Output: Set of clusters $\mathcal{C}$, representing deduplicated records
procedure ANNDEDUPLICATION($\mathcal{E}$, α)
    Initialize set of clusters $\mathcal{C} \leftarrow \emptyset$
    Mark all records in $\mathcal{E}$ as unvisited
    for each record e_i in $\mathcal{E}$ **do**
        if e_i is unvisited **then**
            Mark e_i as visited
            $\mathcal{N} \leftarrow$ FINDALLNEIGHBORS(e_i, $\mathcal{E}$, α)
            if size of $\mathcal{N} \geq 1$ **then**
                Initialize a new cluster $\mathcal{K}$
                Add e_i to $\mathcal{K}$
                for each e_j in $\mathcal{N}$ **do**
                    Add e_j to $\mathcal{K}$
                    Mark e_j as visited
                Add $\mathcal{K}$ to $\mathcal{C}$
    return $\mathcal{C}$
function FINDALLNEIGHBORS(e_i, $\mathcal{E}$, α)
    Initialize empty list $\mathcal{N}$
    for each record e_j in $\mathcal{E}$ **do**
        if $e_j \neq e_i$ and SIMILARITY(e_i, e_j) $\geq \alpha$ **then**
            Add e_j to $\mathcal{N}$
    return $\mathcal{N}$

Appendix B

Examples

B.1 Convolution-Based String Matching

Example (Jaro Similarity)**.** *To demonstrate the Jaro similarity, consider the strings* $S1$ = *"MARTHA" and* $S2$ = *"MARHTA". The matching characters are* $m = 6$*, as all characters in* $S1$ *match with those in* $S2$*, albeit in a slightly different order. The half number of transpositions* t *(where a transposition is a pair of matching characters in a different sequence between* $S1$ *and* $S2$*) is calculated as 1, since two characters ('R' and 'H') are out of order. Thus, the Jaro similarity score* s_J *can be calculated as follows:*

$$s_J(S1, S2) = \frac{1}{3}\left(\frac{m}{|S1|} + \frac{m}{|S2|} + \frac{m-t}{m}\right) \tag{B.1}$$

$$= \frac{1}{3}\left(\frac{6}{6} + \frac{6}{6} + \frac{6-1}{6}\right) \tag{B.2}$$

$$= 0.944 \tag{B.3}$$

Example (Violation of the Triangle Inequality by the Jaro Distance Metric [243])**.** *The Jaro distance metric is not compliant with the triangle inequality, a fundamental property required For metric spaces. For any three elements* $S1$*,* $S2$*, and* $S3$*, the triangle inequality is defined as:*

$$d(S1, S3) \leq d(S1, S2) + d(S2, S3) \tag{B.4}$$

However, consider strings $S1$ = *"ab",* $S2$ = *"cb", and* $S3$ = *"cd". The calculation of the Jaro distance between these strings is as follows:*

The Jaro distance, d_J*, is calculated using the Formula:*

$$d_J(S1, S2) = 1 - s_{Jaro}(S1, S2), \tag{B.5}$$

For $S1 =$ "ab" and $S2 =$ "cb", there is one matching character ('b') with no transpositions ($m = 1$, $t = 0$), leading to:

$$d_J(S1, S2) = 1 - \frac{1}{3}\left(\frac{1}{2} + \frac{1}{2} + 1\right) = \frac{1}{3}. \tag{B.6}$$

Similarly, For $S2 =$ "cb" and $S3 =$ "cd", there is one matching character ('c') with no transpositions ($m = 1$, $t = 0$), hence:

$$d_J(S2, S3) = \frac{1}{3}. \tag{B.7}$$

For $S1 =$ "ab" and $S3 =$ "cd", there are no matching characters ($m = 0$), so:

$$d_J(S1, S3) = 1. \tag{B.8}$$

This results in:

$$d_J(S1, S3) = 1 \not\leq d_J(S1, S2) + d_J(S2, S3) = \frac{1}{3} + \frac{1}{3} = \frac{2}{3}, \tag{B.9}$$

demonstrating the failure of the Jaro distance to comply with the triangle inequality.

Example (Convolutional Jaro Similarity (ConvJ))**.** *Consider the strings $S1 =$ "MARTHA" and $S2 =$ "MARHTA" for calculating Convolutional Jaro similarity with $\sigma = 2$ and $w = 7$.*

Given the matching window determined by $w = 7$, all characters are within this range due to the strings' lengths. The Gaussian weight for each character position difference (distance d) is calculated using $G(i, j) = \exp\left(-\frac{d^2}{2\sigma^2}\right)$, with $\sigma = 2$.

For simplicity, assume the Gaussian weights for matching characters (ignoring character order) result in a sum of weights $M_w = 5.8$ (a hypothetical value for illustrative purposes, reflecting the sum of Gaussian weights for matched characters). Assuming no transpositions for a direct match scenario, the ConvJ score is computed as:

$$s_J(S1, S2) = \frac{1}{3}\left(\frac{M_w}{|S1|} + \frac{M_w}{|S2|} + \frac{M_w}{M_w}\right) \tag{B.10}$$

$$= \frac{1}{3}\left(\frac{5.8}{6} + \frac{5.8}{6} + 1\right) \tag{B.11}$$

$$\approx 0.967 \tag{B.12}$$

This score is slightly adjusted due to the convolutional matching process, illustrating the nuanced similarity assessment provided by ConvJ.

Example (Jaro-Winkler Similarity)**.** *To illustrate the Jaro-Winkler similarity, we examine the strings* $S1 =$ *"DWAYNE" and* $S2 =$ *"DUANE".*

First, calculate the Jaro similarity as above. Assuming $m = 4$ *(matching characters excluding transpositions) and* $t = 1$ *(the 'W' and 'U' are transposed), the Jaro score is:*

$$s_J(S1, S2) = \frac{1}{3}\left(\frac{4}{6} + \frac{4}{5} + \frac{4-1}{4}\right) \approx 0.822 \tag{B.13}$$

The common prefix length l *is 1 (for 'D'), and with* $p = 0.1$ *(the standard scaling factor), the Jaro-Winkler score* s_{JW} *becomes:*

$$\begin{aligned} s_{JW}(S1, S2) &= s_J(S1, S2) + l \cdot p \cdot (1 - s_J(S1, S2)) && \text{(B.14)} \\ &= 0.822 + 1 \cdot 0.1 \cdot (1 - 0.822) && \text{(B.15)} \\ &= 0.822 + 0.018 && \text{(B.16)} \\ &= 0.840 && \text{(B.17)} \end{aligned}$$

This example illustrates how the Jaro-Winkler similarity provides a slight increase over the Jaro similarity for strings with a common prefix, emphasizing the importance of initial characters in certain contexts.

Example (Convolutional Jaro-Winkler (ConvJW))**.** *Consider the strings* $S_1 =$ *"DWAYNE" and* $S_2 =$ *"DUANE", with* $\sigma = 2$ *and decay rate* $\alpha = 0.1$*, where* $w = 7$ *is determined by* σ*. In this ConvJW approach, exponential decay is used instead of the common prefix length for similarity adjustments.*

First, compute the ConvJ similarity with Gaussian-weighted matches. For illustration, let's assume a hypothetical sum of Gaussian-weighted matches $M_w = 4.5$*.*

Unlike the traditional Jaro-Winkler, which applies a prefix scaling factor, the ConvJW similarity here incorporates exponential decay for weighting character matches. This emphasizes the importance of matching characters closer to the beginning of the strings. Assuming the ConvJ similarity score (without the exponential decay adjustment) is approximately 0.822*:*

The adjusted similarity $s_{JW}(S_1, S_2)$ *is then calculated by considering the exponential decay over the indices of matching characters, already factored into* M_w*. Thus, the similarity score remains* 0.822 *in this simplified example, showcasing the effect of the Gaussian and exponential decay directly within the convolutional matching process without an explicit formula here for the adjustment (as it's inherently part of the* M_w *calculation).*

Author's Publications

[OR-1] O. Rozinek and J. Mareš, "The duality of similarity and metric spaces," *Applied Sciences*, vol. 11, no. 4, 2021, ISSN: 2076-3417. DOI: 10.3390/app11041910. [Online]. Available: https://www.mdpi.com/2076-3417/11/4/1910.

[OR-2] O. Rozinek and P. Dolezel, "Ecg hearbeat classification based on multi-scale convolutional neural networks," in *International Work-Conference on Artificial Neural Networks*, Springer, 2023, pp. 352–363.

[OR-3] O. Rozinek and M. Borkovcova, "A novel approach to regression: Exploring the similarity space with ordinary least squares on database records," in *34th Conference of Open Innovations Association (FRUCT)*, IEEE, 2023.

[OR-4] O. Rozinek and M. Borkovcova, "A novel regression approach: Analyzing textual data in similarity space," in *35th Conference of Open Innovations Association (FRUCT)*, IEEE, 2024.

[OR-5] O. Rozinek and J. Mares, "Fast and precise convolutional jaro and jaro-winkler similarity," in *35th Conference of Open Innovations Association (FRUCT)*, IEEE, 2024.

[OR-6] O. Rozinek, J. Marek, J. Panuš, and J. Mareš, "Real-time fuzzy record matching similarity metric and optimal q-gram filter," Manuscript under review for Journal of the ACM, Manuscript ID JACM-00044-2024, Feb. 2024.

[OR-7] O. Rozinek, M. Borkovcova, and J. Mares, "Bipartitejoin: Optimal similarity join for fuzzy bipartite matching," in *World Conference on Information Systems and Technologies*, Springer, 2024.

[OR-8] O. Rozinek, M. Borkovcova, and J. Mares, "Scalable similarity joins for fast and accurate record deduplication in big data," in *World Conference on Information Systems and Technologies*, Springer, 2024.

[OR-9] O. Rozinek and M. Borkovcova, "Theorems for boyd–wong contraction mappings on similarity spaces," *Mathematics*, vol. 11, no. 20, p. 4359, 2023.

References

[1] W. Kim, B.-J. Choi, E.-K. Hong, S.-K. Kim, and D. Lee, "A taxonomy of dirty data," *Data mining and knowledge discovery*, vol. 7, pp. 81–99, 2003.

[2] C. C. H. Elzinga and M. M. Studer, "Normalization of distance and similarity in sequence analysis," *Sequence Analysis and Related Methods (LaCOSA II)*, p. 445, 2016.

[3] W. A. Sutherland, *Introduction to metric and topological spaces*. Oxford University Press, 2009.

[4] E. Alhajjar and C. Lefèvre, "On the similarity metric," *Mathematica Militaris*, vol. 24, no. 1, p. 4, 2019.

[5] M. M. Deza and E. Deza, "Encyclopedia of distances," in *Encyclopedia of distances*, 4th ed. Springer, 2016, pp. 1–583.

[6] S. Chen, B. Ma, and K. Zhang, "On the similarity metric and the distance metric," *Theoretical Computer Science*, vol. 410, no. 24-25, pp. 2365–2376, 2009.

[7] J. Muscat, *Functional analysis: an introduction to metric spaces, Hilbert spaces, and Banach algebras*. Springer, 2014.

[8] C. Alabiso and I. Weiss, *Primer on Hilbert Space Theory*. Springer, 2016.

[9] D. J. Garling, *A Course in Mathematical Analysis: Volume 2, Metric and Topological Spaces, Functions of a Vector Variable*. Cambridge University Press, 2014.

[10] M. Searcóid, *Metric Spaces*. Springer, 2007.

[11] V. A. Zorich and O. Paniagua, *Mathematical analysis II*. Springer, 2016, vol. 220.

[12] D. Lin, "An information-theoretic definition of similarity," in *Proceedings of the 15th International Conference on Machine Learning*, 1998, pp. 296–304.

[13] B. Ma and K. Zhang, "The similarity metric and the distance metric," *Proceedings of the 6th Atlantic Symposium on Computational Biology and GenomeInformatics*, pp. 1239–1242, 2005.

[14] R. N. Shepard, "Toward a universal law of generalization for psychological science," *Science*, vol. 237, no. 4820, pp. 1317–1323, 1987.

[15] P. Gardenfors, *Conceptual spaces: The geometry of thought*. MIT press, 2004.

[16] I. Change, "Climate change 2007: The physical science basis," *Agenda*, vol. 6, no. 07, p. 333, 2007.

[17] C. Huntingford, E. S. Jeffers, M. B. Bonsall, H. M. Christensen, T. Lees, and H. Yang, "Machine learning and artificial intelligence to aid climate change research and preparedness," *Environmental Research Letters*, vol. 14, no. 12, p. 124 007, 2019.

[18] C. Tebaldi and R. Knutti, "The use of the multi-model ensemble in probabilistic climate projections," *Philosophical transactions of the royal society A: mathematical, physical and engineering sciences*, vol. 365, no. 1857, pp. 2053–2075, 2007.

[19] B. B. Mandelbrot, "How long is the coast of britain? statistical self-similarity and fractional dimension," *Science*, vol. 156, no. 3775, pp. 636–638, 1967, In this groundbreaking article, Mandelbrot applies the concept of fractal geometry to quantify the complex, self-similar structure of coastlines, introducing a new perspective on measuring natural forms.

[20] B. B. Mandelbrot, *The Fractal Geometry of Nature*. W. H. Freeman and Company, 1983, This seminal work by Mandelbrot explores the concept of fractals and their application to natural phenomena, laying the foundation for the field of fractal geometry.

[21] S. Chen, B. ma, and K. Zhang, "The normalized similarity metric and its applications," Dec. 2007, pp. 172–180, ISBN: 978-0-7695-3031-4. DOI: 10.1109/BIBM.2007.12.

[22] M. Pagel, Q. D. Atkinson, and A. Meade, "Frequency of word-use predicts rates of lexical evolution throughout indo-european history," *Nature*, vol. 449, no. 7163, pp. 717–720, 2007.

[23] L. Steels, "Modeling the cultural evolution of language," *Physics of life reviews*, vol. 8, no. 4, pp. 339–356, 2011.

[24] R. Bouckaert, P. Lemey, M. Dunn, *et al.*, "Mapping the origins and expansion of the indo-european language family," *Science*, vol. 337, no. 6097, pp. 957–960, 2012.

[25] P. Willett, J. M. Barnard, and G. M. Downs, "Chemical similarity searching," *Journal of chemical information and computer sciences*, vol. 38, no. 6, pp. 983–996, 1998.

[26] A. Bender and R. C. Glen, "Molecular similarity: A key technique in molecular informatics," *Organic & biomolecular chemistry*, vol. 2, no. 22, pp. 3204–3218, 2004.

[27] D. Rogers and M. Hahn, "Extended-connectivity fingerprints," *Journal of chemical information and modeling*, vol. 50, no. 5, pp. 742–754, 2010.

[28] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 113–120.

[29] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proceedings of the fourth ACM international conference on Web search and data mining*, 2011, pp. 177–186.

[30] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive data sets*. Cambridge university press, 2020.

[31] K. M. Borgwardt, C. S. Ong, S. Schönauer, S. Vishwanathan, A. J. Smola, and H.-P. Kriegel, "Protein function prediction via graph kernels," *Bioinformatics*, vol. 21, no. suppl_1, pp. i47–i56, 2005.

[32] G. E. Hinton, "Distributed representations," 1984.

[33] L. W. Barsalou, "Perceptual symbol systems," *Behavioral and brain sciences*, vol. 22, no. 4, pp. 577–660, 1999.

[34] J. Pearl *et al.*, "Models, reasoning and inference," *Cambridge, UK: CambridgeUniversityPress*, vol. 19, no. 2, p. 3, 2000.

[35] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman, "Building machines that learn and think like people," *Behavioral and brain sciences*, vol. 40, e253, 2017.

[36] S. Strogatz, "Sync: The emerging science of spontaneous order," 2004.

[37] K. Börner, S. Sanyal, A. Vespignani, *et al.*, "Network science," *Annu. rev. inf. sci. technol.*, vol. 41, no. 1, pp. 537–607, 2007.

[38] S. Page, *The difference: How the power of diversity creates better groups, firms, schools, and societies-new edition*. Princeton University Press, 2008.

[39] R. Sun, *The Cambridge handbook of computational psychology*. Cambridge University Press, 2008.

[40] S. Zeki, "Inner vision: An exploration of art and the brain," 2002.

[41] H. Leder, B. Belke, A. Oeberst, and D. Augustin, "A model of aesthetic appreciation and aesthetic judgments," *British journal of psychology*, vol. 95, no. 4, pp. 489–508, 2004.

[42] D. Dutton, *The art instinct: Beauty, pleasure, & human evolution*. Oxford University Press, USA, 2009.

[43] U. Kreplin and S. H. Fairclough, "Effects of self-directed and other-directed introspection and emotional valence on activation of the rostral prefrontal cortex during aesthetic experience," *Neuropsychologia*, vol. 71, pp. 38–45, 2015.

[44] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*, 2017.

[45] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[46] W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K.-R. Müller, *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer Nature, 2019, vol. 11700.

[47] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature machine intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[48] P. Prusinkiewicz and A. Lindenmayer, *The Algorithmic Beauty of Plants*. Springer-Verlag, 1990, This book discusses the application of algorithmic and fractal principles to model the growth and form of plants, illustrating how natural structures can be understood through mathematical concepts.

[49] J. A. Adam, *Mathematics in Nature: Modeling Patterns in the Natural World*. Princeton University Press, 2002.

[50] I. Rodriguez-Iturbe and A. Rinaldo, *Fractal River Basins: Chance and Self-Organization*. Cambridge University Press, 1997.

[51] S. Lovejoy, "Area-perimeter relation for rain and cloud areas," *Science*, vol. 216, no. 4542, pp. 185–187, 1982.

[52] K. G. Libbrecht, "The physics of snow crystals," *Reports on Progress in Physics*, vol. 68, no. 4, pp. 855–895, 2005.

[53] M. Stevens and S. Merilaita, *Animal Camouflage: Mechanisms and Function*. Cambridge University Press, 2009.

[54] A. L. Goldberger, D. R. Rigney, and B. J. West, "Chaos and fractals in human physiology," *Scientific American*, vol. 262, no. 2, pp. 42–49, 1990.

[55] P. J. E. Peebles, *The Large-Scale Structure of the Universe*. Princeton University Press, 1980.

[56] D. Bolster, R. E. Hershberger, and R. J. Donnelly, "Dynamic similarity, the dimensionless science," *Physics Today*, vol. 64, no. 9, pp. 42–47, 2011.

[57] R. Djebali, F. Mebarek-Oudina, and C. Rajashekhar, "Similarity solution analysis of dynamic and thermal boundary layers: Further formulation along a vertical flat plate," *Physica Scripta*, vol. 96, no. 8, p. 085 206, 2021.

[58] P. G. Lemarié-Rieusset, *The Navier-Stokes problem in the 21st century*. CRC press, 2018.

[59] A. Dmitrenko, "Reynolds analogy based on the theory of stochastic equations and equivalence of measures," *Journal of Engineering Physics and Thermophysics*, vol. 94, pp. 186–193, 2021.

[60] W. Tang and S.-M. Ngai, "Heat equations defined by self-similar measures with overlaps," *Fractals*, vol. 30, no. 03, p. 2 250 073, 2022.

[61] W. Song, M. Li, Y. Li, C. Cattani, and C.-H. Chi, "Fractional brownian motion: Difference iterative forecasting models," *Chaos, Solitons & Fractals*, vol. 123, pp. 347–355, 2019.

[62] Y. Mishura, K. Ralchenko, M. Zili, and E. Zougar, "Fractional stochastic heat equation with piecewise constant coefficients," *Stochastics and Dynamics*, vol. 21, no. 01, p. 2 150 002, 2021.

[63] P. Kriz and B. Maslowski, "Central limit theorems and minimum-contrast estimators for linear stochastic evolution equations," *Stochastics*, vol. 91, no. 8, pp. 1109–1140, 2019.

[64] P. Kriz and J. Šnupárková, "Pathwise least-squares estimator for linear spdes with additive fractional noise," *Electronic Journal of Statistics*, vol. 16, no. 1, pp. 1561–1594, 2022.

[65] P. Bultinck, X. Gironés, and R. Carbó-Dorcaz, "Molecular quantum similarity: Theory and applications," *Reviews in computational chemistry*, vol. 21, pp. 127–207, 2005.

[66] P. Ghosh and D. Nath, "Generalized quantum similarity index: An application to pseudoharmonic oscillator with isospectral potentials in 3d," *International Journal of Quantum Chemistry*, vol. 121, no. 5, e26517, 2021.

[67] D. Bajusz, A. Rácz, and K. Héberger, "Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?" *Journal of cheminformatics*, vol. 7, no. 1, pp. 1–13, 2015.

[68] A. Cereto-Massagué, M. J. Ojeda, C. Valls, M. Mulero, S. Garcia-Vallvé, and G. Pujadas, "Molecular fingerprint similarity search in virtual screening," *Methods*, vol. 71, pp. 58–63, 2015.

[69] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of molecular biology*, vol. 48, no. 3, pp. 443–453, 1970.

[70] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of molecular biology*, vol. 147, no. 1, pp. 195–197, 1981.

[71] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology*, vol. 215, no. 3, pp. 403–410, 1990.

[72] W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison," *Proceedings of the National Academy of Sciences*, vol. 85, no. 8, pp. 2444–2448, 1988.

[73] J. Felsenstein, "Evolutionary trees from dna sequences: A maximum likelihood approach," *Journal of Molecular Evolution*, vol. 17, no. 6, pp. 368–376, 1981.

[74] J. P. Huelsenbeck and F. Ronquist, "Bayesian inference of phylogeny and its impact on evolutionary biology," *Science*, vol. 294, no. 5550, pp. 2310–2314, 2001.

[75] A. S. Monin and A. M. Obukhov, "Basic laws of turbulent mixing in the surface layer of the atmosphere," *Contributions of the Geophysical Institute of the Slovak Academy of Sciences*, vol. 24, pp. 163–187, 1954.

[76] V. T. Chow, *Open-channel Hydraulics*. McGraw-Hill, 1959.

[77] C. H. Scholz, "The mechanics of earthquakes and faulting," *Cambridge University Press*, 2002.

[78] A. Chao, R. L. Chazdon, R. K. Colwell, and T.-J. Shen, "Abundance-based similarity indices and their estimation when there are unseen species in samples," *Biometrics*, vol. 62, no. 2, pp. 361–371, 2006.

[79] P. J. Somerfield, "Identification of the bray-curtis similarity index: Comment on yoshioka (2008)," *Marine Ecology Progress Series*, vol. 372, pp. 303–306, 2008.

[80] A. E. Magurran, *Ecological Diversity and Its Measurement*. Princeton University Press, 1988.

[81] R. H. Whittaker, "Evolution and measurement of species diversity," *Taxon*, vol. 21, no. 2/3, pp. 213–251, 1972.

[82] M. G. Turner and R. H. Gardner, *Landscape Ecology in Theory and Practice: Pattern and Process*. Springer, 2005.

[83] R. T. Forman, *Land Mosaics: The Ecology of Landscapes and Regions*. Cambridge University Press, 1995.

[84] L. Jost, "Entropy and diversity," *Oikos*, vol. 113, no. 2, pp. 363–375, 2006.

[85] J. D. Anderson, *Fundamentals of Aerodynamics*, 3rd ed. McGraw-Hill, 1999.

[86] H. Chanson, "The hydraulics of open channel flow: An introduction," *Butterworth-Heinemann*, 2004.

[87] M. H. Ray, M. Mongiardini, and C. Plaxico, "Quantitative methods for assessing similarity between computational results and full-scale crash tests," in *Proc. 91th Annu. Meeting Transp. Res. Board*, 2012, pp. 1–21.

[88] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[89] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[90] R. O. Duda, P. E. Hart, *et al.*, *Pattern classification*. John Wiley & Sons, 2006.

[91] S. Kosub, "A note on the triangle inequality for the jaccard distance," *Pattern Recognition Letters*, vol. 120, pp. 36–38, 2019.

[92] M. J. Greenberg, *Euclidean and non-Euclidean geometries: Development and history*. Macmillan, 1993.

[93] W. Rudin, *Functional Analysis*. McGraw-Hill, 1991.

[94] S. Axler, *Linear algebra done right*. Springer Nature, 2023.

[95] J. R. Munkres, *Topology*. Pearson, 2000.

[96] W. Rudin *et al.*, *Principles of mathematical analysis*. McGraw-hill New York, 1976, vol. 3.

[97] J. M. Lee, *Introduction to Smooth Manifolds*. Springer, 2012.

[98] M. M. Fréchet, "Sur quelques points du calcul fonctionnel," *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, vol. 22, no. 1, pp. 1–72, 1906.

[99] F. Hausdorff, *Grundzüge der mengenlehre*. von Veit, 1914, vol. 7.

[100] S. G. Matthews, "Partial metric spaces," University of Warwick. Department of Computer Science, Tech. Rep., 1992.

[101] S. G. Matthews, "The topology of partial metric spaces," University of Warwick. Department of Computer Science, Tech. Rep., 1992.

[102] S. G. Matthews, "Partial metric topology," *Annals of the New York Academy of Sciences*, vol. 728, no. 1, pp. 183–197, 1994.

[103] S. Oltra and O. Valero, "Banach's fixed point theorem for partial metric spaces," *Rend. Istid. Math. Univ. Trieste*, vol. 36, no. 1-2, pp. 17–26, 2004.

[104] M. Bukatin, R. Kopperman, S. Matthews, and H. Pajoohesh, "Partial metric spaces," *The American Mathematical Monthly*, vol. 116, no. 8, pp. 708–718, 2009.

[105] S. Romaguera, "Fixed point theorems for generalized contractions on partial metric spaces," *Topology and its Applications*, vol. 159, no. 1, pp. 194–199, 2012.

[106] S. V. Znamenskij, "From similarity to distance: Axiom set, monotonic transformations and metric determinacy," 2018.

[107] S. T. Wierzchoń and M. A. Kłopotek, *Modern algorithms of cluster analysis*. Springer, 2018, vol. 34.

[108] S. Almazel, Q. H. Ansari, and M. A. Khamsi, *Topics in Fixed Point Theory*. Berlin: Springer-Verlag, 2014.

[109] P. R. Halmos, *Measure theory*. Springer, 2013, vol. 18.

[110] S. Axler, *Measure, integration & real analysis*. Springer Nature, 2022. [Online]. Available: `https://measure.axler.net/`.

[111] L. Yujian and L. Bo, "A normalized levenshtein distance metric," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1091–1095, 2007.

[112] J. B. Conway, *A course in abstract analysis*. American Mathematical Soc., 2012, vol. 141.

[113] J. Bell, "The symmetric difference metric," *Department of Mathematics, University of Toronto*, 2015.

[114] M. Jourlin, *Logarithmic Image Processing: Theory and Applications*. Academic Press, 2016.

[115] J.-C. Bourin and F. Hiai, "Norm and anti-norm inequalities for positive semi-definite matrices," *International Journal of Mathematics*, vol. 22, no. 08, pp. 1121–1138, 2011.

[116] M. Moszyńska and W.-D. Richter, "Reverse triangle inequality. antinorms and semi-antinorms," *Studia Scientiarum Mathematicarum Hungarica*, vol. 49, no. 1, pp. 120–138, 2012.

[117] N. Guglielmi and M. Zennaro, "An antinorm theory for sets of matrices: Bounds and approximations to the lower spectral radius," *Linear Algebra and its Applications*, vol. 607, pp. 89–117, 2020.

[118] A. Podobryaev, "Sub-lorentzian extremals defined by an antinorm," *arXiv preprint arXiv:2402.04687*, 2024.

[119] W. Rudin, *Real and Complex Analysis*, 3rd ed. New York: McGraw-Hill, 1987.

[120] E. Marczewski and H. Steinhaus, "On a certain distance of sets and the corresponding distance of functions," in *Colloquium Mathematicum*, Instytut Matematyczny Polskiej Akademii Nauk, vol. 6, 1958, pp. 319–327.

[121] W. Wu, B. Li, L. Chen, C. Zhang, and S. Y. Philip, "Improved consistent weighted sampling revisited," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 2332–2345, 2018.

[122] P. Jokinen and E. Ukkonen, "Two algorithms for approxmate string matching in static texts," in *International Symposium on Mathematical Foundations of Computer Science*, Springer, 1991, pp. 240–248.

[123] V. I. Levenshtein *et al.*, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet physics doklady*, Soviet Union, vol. 10, 1966, pp. 707–710.

[124] R. A. Wagner and M. J. Fischer, "The string-to-string correction problem," *Journal of the ACM (JACM)*, vol. 21, no. 1, pp. 168–173, 1974.

[125] S. Sagiroglu and D. Sinanc, "Big data: A review," in *2013 international conference on collaboration technologies and systems (CTS)*, IEEE, 2013, pp. 42–47.

[126] R. Feldman and J. Sanger, *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge university press, 2007.

[127] S. M. Stigler, *The history of statistics: The measurement of uncertainty before 1900*. Harvard University Press, 1990.

[128] A.-M. Legendre, "Nouvelles méthodes pour la détermination des orbites des comètes," *Imprimerie Impériale*, 1805.

[129] C. F. Gauss, "Theoria combinationis observationum erroribus minimis obnoxiae," *Werke*, vol. 5, pp. 1–52, 1821.

[130] W. W. Cohen, P. Ravikumar, S. E. Fienberg, *et al.*, "A comparison of string distance metrics for name-matching tasks.," in *IIWeb*, vol. 3, 2003, pp. 73–78.

[131] M. Zhang, J. Tang, and X. Zhang, "Text data processing and analysis: A brief survey," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 2065–2066.

[132] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[133] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

[134] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[135] A. N. Tikhonov and V. Y. Arsenin, "Solution of incorrectly formulated problems and the regularization method," *Soviet Mathematics*, vol. 4, no. 3, pp. 1035–1038, 1963.

[136] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[137] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[138] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[139] P. Agarwal, M. Jleli, and B. Samet, *Fixed Point Theory in Metric Spaces*. Berlin: Springer-Verlag, 2018.

[140] P. Subrahmanyam, *Elementary Fixed Point Theorems*. Berlin: Springer-Verlag, 2014.

[141] H. Pathak, *An Introduction to Nonlinear Analysis and Fixed Point Theory*. Berlin: Springer-Verlag, 2018.

[142] W. Kirk and N. Shahzad, *Fixed Point Theory in Distance Spaces*. Berlin: Springer-Verlag, 2014.

[143] L. E. J. Brouwer, "Über abbildung von mannigfaltigkeiten," *Mathematische Annalen*, vol. 71, pp. 97–115, 1912.

[144] S. Banach, "Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales," *Fundamenta mathematicae*, vol. 3, no. 1, pp. 133–181, 1922.

[145] A. Tarski, "A lattice-theoretical fixpoint theorem and its applications," *Pacific Journal of Mathematics*, vol. 5, no. 2, pp. 285–309, 1955.

[146] E. Rakotch, "A note on contractive mappings," *Proceedings of the American Mathematical Society*, vol. 13, no. 3, pp. 459–465, 1962.

[147] R. Kannan, "Some results on fixed points–ii," *The American Mathematical Monthly*, vol. 76, no. 4, pp. 405–408, 1969.

[148] A. Meir and E. Keeler, "A theorem on contraction mappings," *Journal of Mathematical Analysis and Applications*, vol. 28, no. 2, pp. 326–329, 1969.

[149] D. W. Boyd and J. S. Wong, "On nonlinear contractions," *Proceedings of the American Mathematical Society*, vol. 20, no. 2, pp. 458–464, 1969.

[150] H. Aydi, W. Shatanawi, M. Postolache, Z. Mustafa, N. Tahat, *et al.*, "Theorems for boyd-wong-type contractions in ordered metric spaces," in *Abstract and Applied Analysis*, Hindawi, vol. 2012, 2012.

[151] I. Arandjelović, Z. Kadelburg, and S. Radenović, "Boyd–wong-type common fixed point results in cone metric spaces," *Applied Mathematics and Computation*, vol. 217, no. 17, pp. 7167–7171, 2011.

[152] Z. Kadelburg, S. Radenovic, and S. Shukla, "Boyd-wong and meir-keeler type theorems in generalized metric spaces," *J. Adv. Math. Stud*, vol. 9, no. 1, pp. 83–93, 2016.

[153] F. Nziku and S. Kumar, "Boyd and wong type fixed point theorems in partial metric spaces," *Moroccan Journal of Pure and Applied Analysis*, vol. 5, no. 2, pp. 251–262, 2019.

[154] N. Hussain, Z. Kadelburg, S. Radenović, F. Al-Solamy, *et al.*, "Comparison functions and fixed point results in partial metric spaces," in *Abstract and Applied Analysis*, Hindawi, vol. 2012, 2012.

[155] S. Romaguera and P. Tirado, "The meir–keeler fixed point theorem for quasi-metric spaces and some consequences," *Symmetry*, vol. 11, no. 6, p. 741, 2019.

[156] R. E. Castillo, J. R. Morales, and E. M. Rojas, "Some boyd–wong contraction type mappings in b-metric spaces," *The Journal of Analysis*, vol. 31, no. 2, pp. 911–944, 2023.

[157] P. P. Murthy, Z. Mitrovic, C. P. Dhuri, and S. Radenovic, "The common fixed points in a bipolar metric space," *Gulf Journal of Mathematics*, vol. 12, no. 2, pp. 31–38, 2022.

[158] D. Singh, V. Chauhan, P. Kumam, V. Joshi, and P. Thounthong, "Applications of fixed point results for cyclic boyd–wong type generalized f–ψ-contractions to dynamic programming," *J. Math. Comput. Sci*, vol. 17, pp. 200–215, 2017.

[159] O. Nica, "Existence results for second order three-point boundary value problems," *Differential Equations & Applications*, vol. 4, no. 4, pp. 547–570, 2012.

[160] S. K. Chatterjea, "Fixed point theorems," *C.R. Acad. Bulgare Sci*, vol. 25, pp. 727–730, 1972.

[161] L. B. Ćirić, "A generalization of banach's contraction principle," *Proceedings of the American Mathematical society*, vol. 45, no. 2, pp. 267–273, 1974.

[162] J. Matkowski, "Integrable solutions of functional equations," 1975.

[163] I. Ekeland, "On the variational principle," *Journal of Mathematical Analysis and Applications*, vol. 47, no. 2, pp. 324–353, 1974.

[164] J. Caristi, "Fixed point theorems for mappings satisfying inwardness conditions," *Transactions of the American Mathematical Society*, vol. 215, pp. 241–251, 1976.

[165] B. Rhoades, "Some theorems on weakly contractive maps," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 47, no. 4, pp. 2683–2693, 2001.

[166] T. Suzuki, "A generalized banach contraction principle that characterizes metric completeness," *Proceedings of the American mathematical Society*, vol. 136, no. 5, pp. 1861–1869, 2008.

[167] D. Wardowski, "Fixed points of a new type of contractive mappings in complete metric spaces," *Fixed Point Theory and Applications*, vol. 2012, no. 1, p. 94, 2012.

[168] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[169] R. L. Burden and J. D. Faires, *Numerical Analysis*, 9th ed. Brooks/Cole, Cengage Learning, 2011.

[170] K. Conrad, "The contraction mapping theorem," *Expository paper. University of Connecticut, College of Liberal Arts and Sciences, Department of Mathematics*, 2014.

[171] I. P. Fellegi and A. B. Sunter, "A theory for record linkage," *Journal of the American Statistical Association*, vol. 64, no. 328, pp. 1183–1210, 1969.

[172] N. Koudas, A. Marathe, and D. Srivastava, "Flexible string matching against large databases in practice," in *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, 2004, pp. 1078–1086.

[173] W. E. Winkler, "Matching and record linkage," *Wiley interdisciplinary reviews: Computational statistics*, vol. 6, no. 5, pp. 313–325, 2014.

[174] W. Shen, J. Wang, and J. Han, "Entity linking with a knowledge base: Issues, techniques, and solutions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 2, pp. 443–460, 2014.

[175] S. Chaudhuri, K. Ganjam, V. Ganti, and R. Motwani, "Robust and efficient fuzzy match for online data cleaning," in *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, 2003, pp. 313–324.

[176] V. Christophides, V. Efthymiou, T. Palpanas, G. Papadakis, and K. Stefanidis, "An overview of end-to-end entity resolution for big data," *ACM Computing Surveys (CSUR)*, vol. 53, no. 6, pp. 1–42, 2020.

[177] G. Papadakis, D. Skoutas, E. Thanos, and T. Palpanas, "Blocking and filtering techniques for entity resolution: A survey," *ACM Computing Surveys (CSUR)*, vol. 53, no. 2, pp. 1–42, 2020.

[178] N. Gali, R. Mariescu-Istodor, D. Hostettler, and P. Fränti, "Framework for syntactic string similarity measures," *Expert Systems with Applications*, vol. 129, pp. 169–185, 2019.

[179] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate record detection: A survey," *IEEE Transactions on knowledge and data engineering*, vol. 19, no. 1, pp. 1–16, 2007.

[180] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, and S. Fienberg, "Adaptive name matching in information integration," *IEEE Intelligent Systems*, vol. 18, no. 5, pp. 16–23, 2003.

[181] R. Russell, *Index, us patent 1,261,167*, 1918.

[182] R. C. Russell and U. Russell Index, *Patent 1,435,663;"* 1922.

[183] L. Philips, "Hanging on the metaphone," *Computer Language*, vol. 7, no. 12, pp. 39–43, 1990.

[184] L. Philips, "The double metaphone search algorithm," *C/C++ users journal*, vol. 18, no. 6, pp. 38–43, 2000.

[185] A. J. Lait and B. Randell, "An assessment of name matching algorithms," *Technical Report Series-University of Newcastle Upon Tyne Computing Science*, 1996.

[186] T. Gadd, "Phonix: The algorithm," *Program*, 1990.

[187] R. L. Taft, *Name search techniques*. Bureau of Systems Development, New York State Identification and ..., 1970.

[188] D. Holmes and M. C. McCabe, "Improving precision and recall for soundex retrieval," in *Proceedings. International Conference on Information Technology: Coding and Computing*, IEEE, 2002, pp. 22–26.

[189] P. Christen, "A comparison of personal name matching: Techniques and practical issues," in *Sixth IEEE International Conference on Data Mining-Workshops (ICDMW'06)*, IEEE, 2006, pp. 290–294.

[190] M. A. Jaro, "Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida," *Journal of the American Statistical Association*, vol. 84, no. 406, pp. 414–420, 1989.

[191] W. E. Winkler, "String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage.," 1990.

[192] F. J. Damerau, "A technique for computer detection and correction of spelling errors," *Communications of the ACM*, vol. 7, no. 3, pp. 171–176, 1964.

[193] E. Brill and R. C. Moore, "An improved error model for noisy channel spelling correction," in *Proceedings of the 38th annual meeting of the association for computational linguistics*, 2000, pp. 286–293.

[194] O. Gotoh, "An improved algorithm for matching biological sequences," *Journal of molecular biology*, vol. 162, no. 3, pp. 705–708, 1982.

[195] E. S. Ristad and P. N. Yianilos, "Learning string-edit distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 5, pp. 522–532, 1998.

[196] A. Marzal and E. Vidal, "Computation of normalized edit distance and applications," *IEEE transactions on pattern analysis and machine intelligence*, vol. 15, no. 9, pp. 926–932, 1993.

[197] A. Weigel and F. Fein, "Normalizing the weighted edit distance," in *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3-Conference C: Signal Processing (Cat. No. 94CH3440-5)*, IEEE, vol. 2, 1994, pp. 399–402.

[198] E. Ukkonen, "Approximate string-matching with q-grams and maximal matches," *Theoretical computer science*, vol. 92, no. 1, pp. 191–211, 1992.

[199] P. Christen, "A survey of indexing techniques for scalable record linkage and deduplication," *IEEE transactions on knowledge and data engineering*, vol. 24, no. 9, pp. 1537–1555, 2012.

[200] M. Yu, G. Li, D. Deng, and J. Feng, "String similarity search and join: A survey," *Frontiers of Computer Science*, vol. 10, no. 3, pp. 399–417, 2016.

[201] A. E. Monge, C. Elkan, *et al.*, "The field matching problem: Algorithms and applications.," in *Kdd*, vol. 2, 1996, pp. 267–270.

[202] S. Jimenez, C. Becerra, A. Gelbukh, and F. Gonzalez, "Generalized mongue-elkan method for approximate text string comparison," in *International conference on intelligent text processing and computational linguistics*, Springer, 2009, pp. 559–570.

[203] E. Moreau, F. Yvon, and O. Cappé, "Robust similarity measures for named entities matching," in *COLING 2008*, ACL, 2008, pp. 593–600.

[204] J. Wang, G. Li, and J. Fe, "Fast-join: An efficient method for fuzzy token matching based string similarity join," in *2011 IEEE 27th International Conference on Data Engineering*, IEEE, 2011, pp. 458–469.

[205] J. Wang, G. Li, and J. Feng, "Extending string similarity join to tolerant fuzzy token matching," *ACM Transactions on Database Systems (TODS)*, vol. 39, no. 1, pp. 1–45, 2014.

[206] D. Deng, A. Kim, S. Madden, and M. Stonebraker, "Silkmoth: An efficient method for finding related sets with maximum matching constraints," *arXiv preprint arXiv:1704.04738*, 2017.

[207] A. Gragera and V. Suppakitpaisarn, "Relaxed triangle inequality ratio of the sørensen–dice and tversky indexes," *Theoretical Computer Science*, vol. 718, pp. 37–45, 2018.

[208] J. Wang, C. Lin, and C. Zaniolo, "Mf-join: Efficient fuzzy string similarity join with multi-level filtering," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, IEEE, 2019, pp. 386–397.

[209] M. H. DeGroot, *Optimal statistical decisions*. John Wiley & Sons, 2004.

[210] A. Andoni, M. Deza, A. Gupta, P. Indyk, and S. Raskhodnikova, "Lower bounds for embedding edit distance into normed spaces," 2003.

[211] K. Hosseini, F. Nanni, and M. C. Ardanuy, "Deezymatch: A flexible deep learning approach to fuzzy string matching," in *Proceedings of the 2020 conference on empirical methods in natural language processing: System demonstrations*, 2020, pp. 62–69.

[212] P. Ferragina and U. Scaiella, "Tagme: On-the-fly annotation of short text fragments (by wikipedia entities)," in *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010, pp. 1625–1628.

[213] R. Santos, P. Murrieta-Flores, P. Calado, and B. Martins, "Toponym matching through deep neural networks," *International Journal of Geographical Information Science*, vol. 32, no. 2, pp. 324–348, 2018.

[214] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[215] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[216] L. Qiu, J. Yu, Q. Pu, and C. Xiang, "Knowledge entity learning and representation for ontology matching based on deep neural networks," *Cluster Computing*, vol. 20, pp. 969–977, 2017.

[217] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," *arXiv preprint arXiv:1606.01781*, 2016.

[218] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[219] G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys*, vol. 33, no. 1, pp. 31–88, 2001. DOI: 10.1145/375360.375365.

[220] Y. Wang, J. Qin, and W. Wang, "Efficient approximate entity matching using jaro-winkler distance," in *International conference on web information systems engineering*, Springer, 2017, pp. 231–239.

[221] K. Dreßler and A.-C. Ngonga Ngomo, "On the efficient execution of bounded jaro-winkler distances," *Semantic Web*, vol. 8, no. 2, pp. 185–196, 2017.

[222] P. Pitchandi and M. Balakrishnan, "Document clustering analysis with aid of adaptive jaro winkler with jellyfish search clustering algorithm," *Advances in Engineering Software*, vol. 175, p. 103 322, 2023.

[223] *Oracle database data quality operators documentation*, https://docs.oracle.com/en/database/oracle/oracle-database/23/sqlrf/data-quality-operators.html, Accessed: 2024-03-19, 2023.

[224] *Talend open studio documentation*, https://help.talend.com/r/en-US/8.0/studio-user-guide/defining-matching-key, Accessed: 2024-03-19.

[225] *Ibm infosphere documentation*, https://www.ibm.com/docs/en/ignm/7.0.0?topic=overview-score-type, Accessed: 2024-03-19.

[226] *Informatica data quality documentation*, https://docs.informatica.com/, Accessed: 2024-03-19.

[227] *Apache lucene documentation*, https://lucene.apache.org, Accessed: 2024-03-19.

[228] Rosetta Code, *Jaro similarity*, https://rosettacode.org/wiki/Jaro_similarity, Accessed: 2024-02-28, 2024.

[229] H. Köpcke, A. Thor, and E. Rahm, "Evaluation of entity resolution approaches on real-world match problems," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 484–493, 2010.

[230] DBS Universität Leipzig, *Benchmark datasets for entity resolution*, https://dbs.uni-leipzig.de/en/research/projects/object_matching/benchmark_datasets_for_entity_resolution, Accessed: 2023-02-21, 2023. [Online]. Available: https://dbs.uni-leipzig.de/en/research/projects/object_matching/benchmark_datasets_for_entity_resolution.

[231] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[232] H. W. Kuhn, "Variants of the hungarian method for assignment problems," *Naval research logistics quarterly*, vol. 3, no. 4, pp. 253–258, 1956.

[233] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the society for industrial and applied mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[234] R. Duan and S. Pettie, "Linear-time approximation for maximum weight matching," *Journal of the ACM (JACM)*, vol. 61, no. 1, pp. 1–23, 2014.

[235] J. Edmonds and R. M. Karp, "Theoretical improvements in algorithmic efficiency for network flow problems," *Journal of the ACM (JACM)*, vol. 19, no. 2, pp. 248–264, 1972.

[236] C. Xiao, W. Wang, X. Lin, J. X. Yu, and G. Wang, "Efficient similarity joins for near-duplicate detection," *ACM Transactions on Database Systems (TODS)*, vol. 36, no. 3, pp. 1–41, 2011.

[237] Z. Yang, J. Yu, and M. Kitsuregawa, "Fast algorithms for top-k approximate string matching," in *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[238] K. R. Rasmussen, J. Stoye, and E. W. Myers, "Efficient q-gram filters for finding all ε-matches over a given length," *Journal of Computational Biology*, vol. 13, no. 2, pp. 296–308, 2006.

[239] P. Grzebala and M. Cheatham, "Private record linkage: Comparison of selected techniques for name matching," in *European Semantic Web Conference*, Springer, 2016, pp. 593–606.

[240] H. Sababa and A. Stassopoulou, "A classifier to distinguish between cypriot greek and standard modern greek," in *2018 Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, IEEE, 2018, pp. 251–255.

[241] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *kdd*, vol. 96, 1996, pp. 226–231.

[242] Z. Dafir, Y. Lamari, and S. C. Slaoui, "A survey on parallel clustering algorithms for big data," *Artificial Intelligence Review*, vol. 54, pp. 2411–2443, 2021.

[243] M. P. Van der Loo *et al.*, "The stringdist package for approximate string matching.," *R J.*, vol. 6, no. 1, p. 111, 2014.