International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

"A Review on String-Based Text Similarity Techniques in Computational Analysis "

Navjot Kaur

Submitted: 07/11/2024 **Revised:** 20/12/2024 **Accepted:** 29/12/2024

Abstract: Text similarity between words, sentences, paragraphs or documents has a great significance in all the application of Natural Language Processing (NLP) like information retrieval, word sense disambiguation, machine translation, text summarization etc. In this paper researcher have presented the survey of various string based methods used to find the text similarity. All the methods come under the two broad approaches which are character-based and term-based. These days measuring the text similarity between words, lines or documents plays important role for researches in the fields related to text such as plagiarism detection, machine translation, information retrieval etc.

Keywords: *Text Similarity, String-Based Similarity, Character-Based Similarity, Term-based Similarity*

1. INTRODUCTION

World Wide Web is a highly dynamic, widely used, huge information source with billions of documents and trillions of terabytes of data covering almost every possible topic with numerous data transactions (updations, additions, and modifications) taking place in every second, increasing the volume of this humungous database at an exponential pace. On Web same file is placed at multiple places. Many Researchers or students misuse the data on web documents without customizing or writing authors. Measuring the text similarity between words, sentences or documents plays important role for researches in the fields related to text such as plagiarism detection, machine translation, information retrieval etc

Text similarity between words can be calculated in two ways lexically and semantically. In lexical text similarity measures the character sequencing is used whereas in semantically text similarity is about the meaning closeness.

For Example:

- The dog bites the man
- The man bites the dog

According to the lexical similarity, the above two phrases are almost identical because they have the same word set. For semantic similarity, they are completely different because they have different meanings despite the similarity of the word set.

String-Based and Semantic-Based methods are the two main methods which are used for text similarity measures. String-Based algorithms use the concept of lexical similarity which is further divided in to two main parts: Term-Based and Character-Based approaches[13,14,15]. Measuring Text Similarity plays an important role in various NLP applications like information retrieval, sentiment analysis, text Summarization, Word sense Disambiguation etc. Various case studies have also been carried out by the various researchers to find the text similarity the various methods, tools, software's and applications of semantic similarity. Khuat Thanh Tung et.al in 2015, compared the effectiveness of algorithms to measure the similarity between two documents using a character-based(n-gram) and a term-based algorithm(Dice coefficient).

2. STRING-BASED SIMILARITY MEASURES

String sequences and character composition is string similarity measures. These methods are further divided in to two categories which are Term-Based and Character-Based[15]. Various algorithms of both term based and character based algorithms are discussed in the following section.

Assistant Professor, Department of Computer Science and Engineering

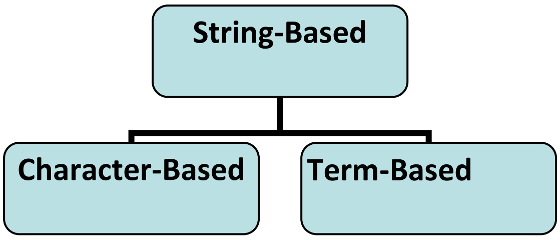


Figure 1: Classification of String-Based Text Similarity Methods

3. CHARACTER-BASED METHODS

Character-based similarity measures are also known as sequence-based measurement. It is also known as Edit-distance based measurement, where edit distance is calculated between two strings of characters [1,2,4,6,15]. Algorithms comes under Character-Based methods are discussed below. Figure 2 shows various algorithms comes under Character-Based Methods.

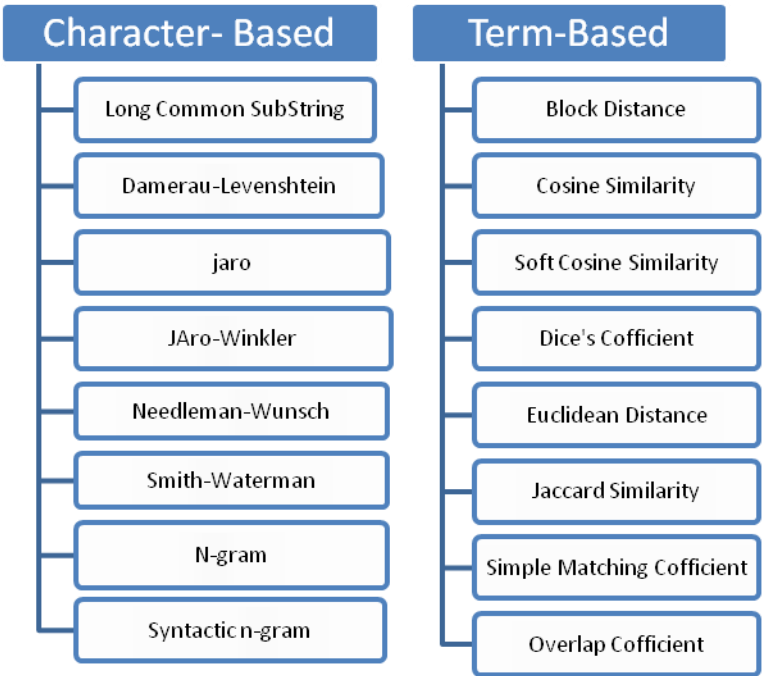


Figure 2: Character-Based and Term-Based Algorithms

3.1 Longest Common SubString (LCS) finds the longest common chain of characters by comparing two strings. For example S1={B,C,D,A,A,C,D} and S2={A,C,D,B,A,C} ,then the common subsequences are {B, C}, {C, D, A, C}, {D, A, C}, {A, A, C}, {A, C}, {C, D}. Among these subsequences, {C, D, A, C} is the longest common subsequence.

3.2 Damerau-Levenshtein algorithm calculates the minimum count of steps which can be used to change input string into to targeted string by using insertion, deletion, substitution or transposition of characters.For example, strings A = "a cat" and B = "an act." The Levenshtein distance for this is 3: to get from A to B requires one addition (the 'n') and two substitutions ('a' to 'c' and 'c' to 'a').

3.3 Jaro measures is designed to compare the short string based on how many number of characters and the order of characters common between two strings. Jaro distance value ranges from 0 to 1 where 1 means strings are equal and 0 means they are not equal [10,15].

$$Jaro\ similarity\ = \begin{cases} 0, \text{ if m=0} \\ \frac{1}{3}\left(\frac{m}{|s1|}+\frac{m}{|s2|}+\frac{m-t}{m}\right), \text{ for m!=0} \end{cases}$$

……………………..(1)

m= number of matching characters but in different order

t= half the number of transpositions.

|s1|= length of string 1

|s2| = length of string s2

$$\left\lfloor \frac{max(|s1|, |s2|)}{2} \right\rfloor - 1$$

……………………………………(2)

If the characters are same and they are not at the distance more than the value calculated in equation 2 then the characters are said to be matching.

For Example S1=”gurnav” and S2=”navgur”, so the maximum to which each character is matched is 1. In this example both string S1 and string S2 have same letters and same length but the order of letters is not same. The number of matching characters that are not in order are 6 and transposition is 3 and the jaro similarity is:

Jaro Similarity = (1/3) * {(6/6) + (6/6) + (6-3)/6 } = 0.833333

3.4 Jaro–Winkler is the extension of Jaro distance which is also used to comare short strings but Jaro-Winkler gives higher similarity for strings that match from the beginning. It uses a pre fixed scale p which gives more suitable rating to strings that have a common prefix up to a defines maximum length l. Jaro-Wrinkler similarity is calculated with the following formula[10,11]:

Sw = Sj + P * L * (1 – Sj)………………………………(3)

In above euation Sj is called jaro similarity, Sw is called jaro-winkler similarity ,P is the scaling factor which is 0.1 by default and last the L is the length of matching prefix which could be at max of four characters. For example S1=”gurnav” and S2=”gunrav” . The length of the matching prefix is 2, sacaling factor is 0.1 and the jaro similarity is 0.944444 and from this jaro-winker similarity will be as follows:

Jaro-Winkler Similarity= 0.944444 + 0.1 * 2 * (1-0.944444) = 0.9555552

3.5 Needleman-Wunsch algorithm is referred to as optimal algorithm which is used in biological sequence comparisons and performs a global sequence alignment. It was developed by Saul B. Needleman and Christian D. Wunsch in 1970. Scoring system use in Needleman-Wunsch is as follows:

- Match score (e.g., +1)
- Mismatch penalty (e.g., -1)
- Gap penalty (e.g., -2)

3.6 Smith-Waterman is also the dynamic algorithm that performs local sequences alignment. In this segment of all possible lengths are compared and optimizes the text similarity.. Because of its cubic computational complexity, it is not used in large scale problem [13]. Use match, mismatch, and gap penalties:

- Match = +2
- Mismatch = -1
- Gap = -2

3.7 N-grams are a fundamental concept used in text similarity, especially in string-based or token-based similarity measures which is about finding the probability distribution over word sequences. N-gram model is a type of language model. An N-gram [11, 13] is a contiguous sequence of N items (usually characters or words) extracted from a piece of text. In text similarity, character-level n-grams are most commonly used, which compares the n-grams from each character or word in two strings.

$$Ngrams_K = X - (N - 1)$$

....(4)

The number of N-gram for any sentence k would be according to the equation (4) shown above. In this equation X is the Number of words in a given sentence K. For example S1= “Diana looked at the strange piglets”.

If N=3 and the N-gram would be:

- Diana looked at
- looked at the
- at the strange
- the strange piglets

3.8 Syntactic N-gram: Syntactic n-grams are n-grams defined by paths in syntactic dependency or constituent trees rather than the linear structure of the text.

4. TERM BASED METHODS

Term-based techniques in text similarity compare texts based on terms (usually words or tokens) rather than characters. The drawbacks of character-based are addressed by the term-based similarity method, when you work on large strings [1,3,5,7].

4.1 Block Distance is a distance metric used to measure the difference between two vectors is also known with some other names like boxcar distance, taxicab distance, absolute value distance, Manhattan distance, snake distance, L1 distance or city block distance[13]. The Block distance find the distance between two points using the following formula:

$$d_1(p,q) = \|p-q\|_1 = \sum_{i=1}^{n} |p_i - q_i|.$$

…………………..(5)

For example

Text A: {"apple": 2, "banana": 1}

Text B: {"apple": 1, "banana": 2}

Block Distance = |2–1| + |1–2| = **2**

4.2 Cosine similarity is a metric used to measure the text similarity between two text documents[12]. It measures the cosine of the angle between two term frequency vectors projected in a multi-dimensional space. The smaller the angle, higher the cosine similarity .Given vectors the attribute A and B, the cosine similarity is calculated as:

$$similarity = cos(\theta) = \frac{A.B}{|A||B|}$$

……………………………….(6)

Consider these two documents:

- Doc1: "I like apples"
- Doc2: "I like oranges"

Using **Bag of Words**, the vocabulary is: ["I", "like", "apples", "oranges"]

Vectors:

- Doc1 = [1, 1, 1, 0]
- Doc2 = [1, 1, 0, 1]

Cosine similarity =

$$\frac{(1\times1+1\times1+1\times0+0\times1)}{\sqrt{1^2+1^2+1^2+0^2}\times\sqrt{1^2+1^2+0^2+1^2}} = \frac{2}{\sqrt{3}\times\sqrt{3}} = \frac{2}{3} \approx 0.667$$

The cosine similarity can be used when you want to compare two documents,sentences. It is also used in classification and clustering.

4.3 Soft Cosine Similarity is an extension of cosine similarity that accounts for semantic similarity between features (words).This technique is used when your documents are semantically same. Unlike cosine similarity it consider the similarity of features in VSM (Vector Space Model).Soft cosine similarity is measured as follows:

$$soft\_cos(\theta) = \frac{\sum\limits_{i,j=1}^{n} s_{ij}A_iB_j}{\sqrt{\sum\limits_{i,j=1}^{n} s_{ij}A_iA_j}\sqrt{\sum\limits_{i,j=1}^{n} s_{ij}B_iB_j}},$$

…………………………………..(7)

Cosine similarity assumes all features (words) are independent and orthogonal. For example in cosine similarity the two words "car" and "automobile" are treated as completely different. They are synonyms but in cosine similarity they have low similarity scores[14]. But in case of soft cosine similarity captures the semantic similarity between car and automobile.

4.4 Dice's coefficient also known as Sorensen-Dice (DSC) index is a statistical tool. The following equation shows how to calculate the text similarity using DSC. In this union of two sets is divided with the intersection of two sets and further the result is multiplied by 2 [1,15]**.** It is commonly used in text similarity by comparing sets of words or character n-grams.

$$\mathrm{Dice}(A,B)=\frac{2\cdot|A\cap B|}{|A|+|B|}$$

…………..……………………..……(8)

Where A and B are sets of tokens (e.g., words or character bigrams),

Text A = "night" → Bigrams = {"ni", "ig", "gh", "ht"}

Text B = "nacht" → Bigrams = {"na", "ac", "ch", "ht"}

Common bigram = {"ht"}

$$\mathrm{Dice}=\frac{2\cdot 1}{4+4}=\frac{2}{8}=0.25$$

This technique is commonly used for short text similarity, for name matching or for spell check suggestions. This technique is also used in bioinformatics for example comparing DNA or protein sequences.

4.5 Euclidean distance or L2 distance measures the **straight-line distance** between two vectors in multi-dimensional space. These vectors are typically generated using TF-IDF (Term Frequency-Inverse Document Frequency), Bag of Words (BoW) or Word embeddings (e.g., Word2Vec, GloVe) method. It is calculated using the following formula[1]:

$$d\left(s,t\right)=d\left(t,s\right)=\sqrt{\sum_{i=1}^{n}\left(t_i-p_i\right)^2}.$$

………………………..….(9)

The above equation finds the Euclidean distance between two points s and t

Compare these two texts:

- **Text A**: "I like apples"
- **Text B**: "I like oranges"

If represented as TF-IDF vectors:

A = [0.58, 0.58, 0.58, 0] (word: "I", "like", "apples", "oranges")

B = [0.58, 0.58, 0, 0.58]

$$\text{Euclidean Distance}=\sqrt{(0.58-0.58)^2+(0.58-0.58)^2+(0.58-0)^2+(0-0.58)^2}\approx 0.82$$

This technique is easy to compute but it does not handle the synonyms well and this technique is good for dense and low dimensional data .

4.6 Jaccard similarity is a measure of how dissimilar two sets are. It is calculated using the following formula[7]:

J(S,T) = |S∩T| / |S∪T| …………………………………………….(10)

In equation 10 the ratio of the size of the intersection of S and T to the size of their union.

4.7 Simple Matching Coefficient is a very simple vector based approach which is defined as the ratio of the total number of matching attributes, on which both vectors are non zero to the total number of attributes present . SMC between A and B is calculated as follows:

$$\begin{aligned}\mathrm{SMC}&=\frac{\text{number of matching attributes}}{\text{number of attributes}}\\&=\frac{M_{00}+M_{11}}{M_{00}+M_{01}+M_{10}+M_{11}}\end{aligned}$$

……………………………(11)

where, M_{00} is the total number of attributes where both objects obj1 and obj2 are 0, M_{01} is the total number of attributes where obj1 is 0 and obj2 is 1, M_{10} is the total number of attribute where obj1 is 1 and obj2 is 0 and M_{11} is the total number of attributes where both obj1 and obj2 are 1**.**

4.8 Overlap coefficient is similar to the Dice's coefficient and is also called Szymkiewicz-Simpson coefficient. In this method two strings are considered as completely similar if one is a subset of the other. It measures the overlap between two finite sets.

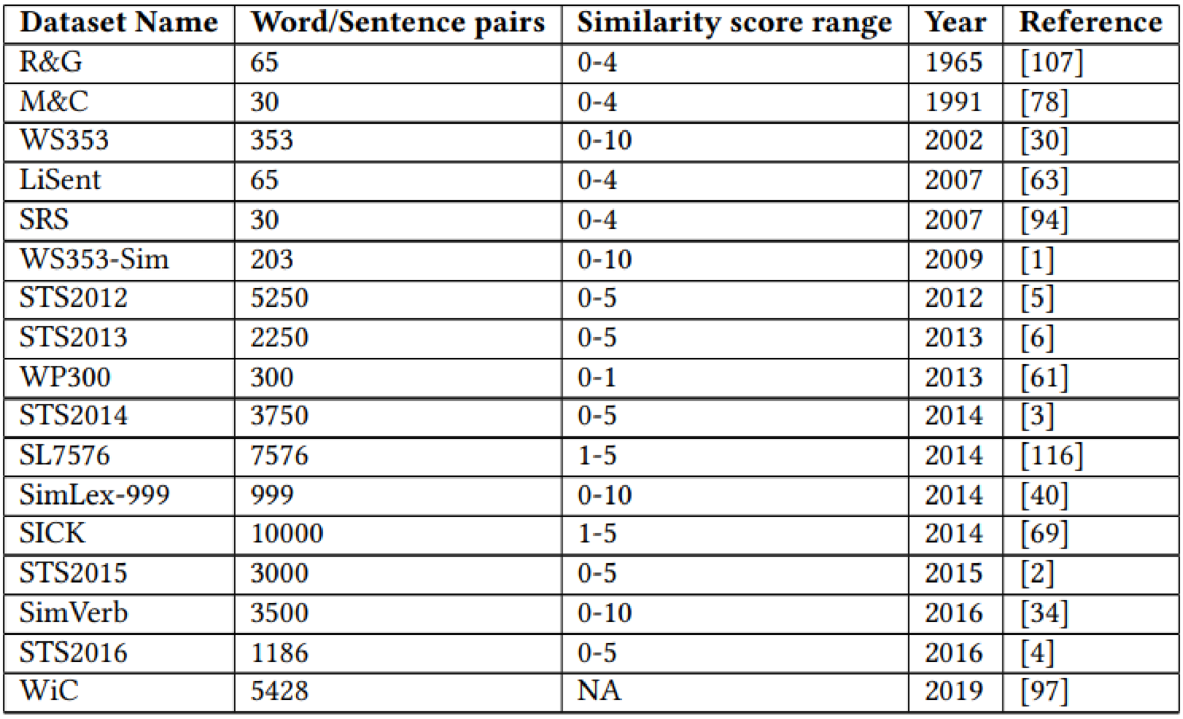
$$overlap(A,B) = \frac{|A \cap B|}{\min(|A|, |B|)}.$$

……………………………..(12)

5 DATESET

In this section the researcher has discussed the most widely used dataset to find the semantic text similarity between two words, strings, paragraph or documents shown in Table 1. The datasets may include word pairs or sentence pairs with associated standard similar.

Table 1.Popular benchmark datasets for Semantic similarity [8]



Dataset Name	Word/Sentence pairs	Similarity score range	Year	Reference
R&G	65	0-4	1965	[107]
M&C	30	0-4	1991	[78]
WS353	353	0-10	2002	[30]
LiSent	65	0-4	2007	[63]
SRS	30	0-4	2007	[94]
WS353-Sim	203	0-10	2009	[1]
STS2012	5250	0-5	2012	[5]
STS2013	2250	0-5	2013	[6]
WP300	300	0-1	2013	[61]
STS2014	3750	0-5	2014	[3]
SL7576	7576	1-5	2014	[116]
SimLex-999	999	0-10	2014	[40]
SICK	10000	1-5	2014	[69]
STS2015	3000	0-5	2015	[2]
SimVerb	3500	0-10	2016	[34]
STS2016	1186	0-5	2016	[4]
WiC	5428	NA	2019	[97]

6 CONCLUSION AND FUTURE SCOPE

The text similarity based methods are categorized in to two main categories they are string based and semantic based. In this survey the only string based methods have been discussed. String based text similarity methods are further of two types they are character-based and term based methods. This paper includes the discussion of eight character-based and eight term-based text similarity measures with example. String-Based measures operate on string sequences and character composition, they are simple and easy to use but they are only used for dissimilarity or distance measures. This survey would serve as a good foundation for researchers who intend to find how the string-based methods are used to find the text similarity.

References

[1] Gomaa, W.H., Fahmy, A.A., "A Survey of Text Similarity Approaches", *International Journal of Computer Applications*, Vol. 68, No. 13,pp: 13–18, 2013.

[2] Alberto, B. , Paolo, R., Eneko A. & Gorka L. , "Plagiarism Detection across Distant Language Pairs", *In Proceedings of the 23rd International Conference on Computational Linguistics*, pp 37–45, 2010.

[3]P. Sitikhu, K. Pahi, P. Thapa and S. Shakya, "A Comparison of Semantic Similarity Methods for Maximum Human Interpretability", *Artificial Intelligence for Transforming Business and Society (AITB)*, Kathmandu, Nepal, pp.1-4, 2019, doi: 10.1109/AITB48515.2019.8947433.

[4] Goutam Majumder, Partha Pakray, Alexander Gelbukh, David Pinto, "Semantic Textual Similarity Methods, Tools, and Applications: A Survey", *Computación y Sistemas*, Vol. 20, No. 4, pp. 647–665, 2016.

[5] N. Shibata, Y. Kajikawa, I. Sakata, "How to measure the semantic similarities between scientific papers and patents in order to discover uncommercialized research fonts: A case study of solar cells", I*n Proceedings of PICMET technology management for global economic growth, Phuket*, pp. 1-6, 2010.

[6] Jiapeng Wang and Yihong Dong , "Measurement of Text Similarity: A Survey" , Information, Vol. 11,No. 421,pp. 1-17,2020.

[7] Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. Bulletin de la Société Vaudoise des Sciences Naturelles 37, 547-579.

[8] Khuat Thanh Tung, Nguyen Duc Hung, Le Thi My Hanh, "A Comparison of Algorithms used to measure the Similarity between two documents ",*International Journal of Advanced Research in Computer Engineering & Technology*, Vol 4, No 4,pp. 117-1121,2015.

[9] Dhivya Chandrasekaran and Vijay Mago, "Evolution of Semantic Similarity - A Survey", *ACM Computing Surveys (CSUR)*, Vol. 54, No. 2,pp. 1-37, February 2020.

[10] Jaro, M. A. (1989). Advances in record linkage methodology as applied to the 1985 census of Tampa Florida, Journal of the American Statistical Society, vol. 84, 406, pp 414-420

[11] Winkler W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage, Proceedings of the Section on Survey Research Methods, American Statistical Association, 354–359.

[12] Saurabh Agarwala, Aniketh Anagawadi, Ram Mohana Reddy Guddeti, "Detecting Semantic Similarity Of Documents Using Natural Language Processing",*Procedia Computer Science*,Vol.189,pp. 128-135,2021, ISSN 1877-0509,https://doi.org/10.1016/j.procs.2021.05.076.

[13] Majumder, Goutam & Pakray, Dr. Partha & Gelbukh, Alexander, "Semantic Textual Similarity Based On Uni-Gram Language Model And Lexical Taxonomy", *International Journal of Computational Linguistics and Applications*, Vol. 20, No. 4, pp. 647–665, 2016.

[14] Prakoso, D.W., Abdi, A. & Amrit, C. Short text similarity measurement methods: a review. *Soft Comput* **25**, pp. 4699–4723 , 2021. https://doi.org/10.1007/s00500-020-05479-2

[15]Web link accessed on16 October 2021: https://www.geeksforgeeks.org/jaro-and-jaro-winkler-similarity/