Regular Language Distance and Entropy*

Austin J. Parker[1], Kelly B. Yancey[2], and Matthew P. Yancey[3]

1 **Institute for Defense Analyses - Center for Computing Sciences, Bowie MD, USA;** ajpark2@super.org
2 **University of Maryland, College Park MD, USA; Institute for Defense Analyses - Center for Computing Sciences, Bowie MD, USA;** kbyancey1@gmail.com
3 **Institute for Defense Analyses - Center for Computing Sciences, Bowie MD, USA;** mpyancey1@gmail.com

Abstract

This paper addresses the problem of determining the distance between two regular languages. It will show how to expand Jaccard distance, which works on finite sets, to potentially-infinite regular languages.

The entropy of a regular language plays a large role in the extension. Much of the paper is spent investigating the entropy of a regular language. This includes addressing issues that have required previous authors to rely on the upper limit of Shannon's traditional formulation of channel capacity, because its limit does not always exist. The paper also includes proposing a new limit based formulation for the entropy of a regular language and proves that formulation to both exist and be equivalent to Shannon's original formulation (when it exists). Additionally, the proposed formulation is shown to equal an analogous but formally quite different notion of topological entropy from Symbolic Dynamics – consequently also showing Shannon's original formulation to be equivalent to topological entropy.

Surprisingly, the natural Jaccard-like entropy distance is trivial in most cases. Instead, the *entropy sum* distance metric is suggested, and shown to be granular in certain situations.

1998 ACM Subject Classification F.4.3 Formal Languages, G.2.2 Graph Theory

Keywords and phrases regular languages, channel capacity, entropy, Jaccard, symbolic dynamics

Digital Object Identifier 10.4230/LIPIcs.MFCS.2017.3

1 Introduction

In this paper we study distances between regular expressions. There are many motivations for this analysis. Activities in bioinformatics, copy-detection [9], and network defense sometimes require large numbers of regular expressions be managed. Metrics aid in indexing and management of those regular expressions [4]. Further, understanding the distance between regular languages requires an investigation of the structure of regular languages that we hope eliminates the need for similar theoretical investigations in the future.

A natural definition of the distance between regular languages L_1 and L_2 containing strings of symbols from Σ is: $\lim_{n\to\infty} \frac{|(L_1 \Delta L_2)\cap\Sigma^n|}{|(L_1\cup L_2)\cap\Sigma^n|}$ (where $L_1 \Delta L_2 = (L_1 \cup L_2) \setminus (L_1 \cap L_2)$ is the symmetric difference). However, the definition has a fundamental flaw: the limit does not always exist. Consider the distance between $(aa)^*$ and a^*. When n is even, the fraction is 0, while when n is odd the fraction is 1. Thus, the limit given above is not well defined for those two languages.

* For a full version see [22].

42nd International Symposium on Mathematical Foundations of Computer Science (MFCS 2017).
Editors: Kim G. Larsen, Hans L. Bodlaender, and Jean-Francois Raskin; Article No. 3; pp. 3:1–3:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This paper addresses that flaw and examines the question of entropy and distance between regular languages in a more general way. A fundamental contribution will be a limit-based distance related to the above that (1) exists, (2) can be computed from the Deterministic Finite Automaton for the associated regular languages, and (3) does not invalidate expectations about the distance between languages.

The core idea is two-fold: (1) to rely on the number of strings *up-to* a given length rather than strings *of* a given length and (2) to use Cesáro averages to smooth out the behavior of the limit. These ideas led us to develop the Cesáro Jaccard distance, which is proven to be well-defined in Theorem 7.

Tied up in this discussion will be the *entropy* of a regular language, which is again a concept whose common definition needs tweaking due to limit-related considerations.

This paper is structured as follows. In Section 2 we discuss related work and define terms that will be used in the paper. Of particular importance is Table 1, which includes all of the distance functions defined in the paper. As the Jaccard distance is a natural entry point into distances between sets, Section 3 will discuss the classical Jaccard distance and how best to extend it to infinite sets. Section 4 will discuss notions of regular language entropy, introducing a new formulation and proving it correct from both a channel capacity and a topological entropy point of view. Section 5 will introduce some distances based on entropy, and show that some of them behave well, while others do not. Finally, Section 6 provides a conclusion and details some potential future work.

2 Background

2.1 Related Work

Chomsky and Miller's seminal paper on regular languages [6] does not address distances between regular languages. It uses Shannon's notion of channel capacity (equation 7 from [6]) for the entropy of a regular language: $h(L) = \lim_{n\to\infty} \frac{\log|L\cap\Sigma^n|}{n}$.

While Shannon says of that limit that "the limit in question will exist as a finite number in most cases of interest" [27], its limit does not always exist for regular languages (consider $(\Sigma^2)^*$). This motivates much of the analysis in this paper. Chomsky and Miller also examine the number of sentences *up to* a given length, foreshadowing some other results in this paper. However, their analysis was based upon an assumption with deeper flaws than that the limit exists. In this paper we address those issues.

Several works since Chomsky and Miller have used this same *of length exactly* n formula to define the entropy of a regular language [3, 9, 17]. These works define entropy as Chomsky and Miller, but add the caveat that they use the upper limit when the limit does not exist. Here we provide foundation for those works by showing the upper limit to be correct (Theorem 13). Further, this paper suggests an equivalent expression for entropy that may be considered more elegant: it is a limit that exists as a finite number for all regular languages which equals the traditional notion of entropy when that limit exists.

Chomsky and Miller's technique was to develop a recursive formula for the number of words accepted by a regular language. That recursive formula comes from the characteristic polynomial of the adjacency matrix for an associated automaton. The eigenvalues of the adjacency matrix describe the growth of the language (we use the same technique, but apply stronger theorems from linear algebra that were discovered several decades after Chomsky and Miller's work). The recursive formula can also be used to develop a generating function to describe the growth of the language (see [25]). Bodirsky, Gärtner, Oertzen, and Schwinghammer [2] used the generating functions to determine the growth of a regular language over alphabet Σ relative to $|\Sigma|^n$, and Kozik [16] used them to determine the growth of a regular language relative to a second regular language. Our approaches share significant details: they relate the growth of a regular language to the poles of its generating function—which are the zeroes of the corresponding recurrence relation—which are the eigenvalues of the associated adjacency matrix. Our technique establishes the "size" of a regular language independent of a reference alphabet or language.

There is work examining distances between *unary* regular languages, or regular languages on the single character alphabet ($|\Sigma| = 1$) [11]. It introduces a definition for Jaccard distance that will appear in this paper: $1 - \lim_{n\to\infty} \frac{|L_1 \cap L_2 \cap (\bigcup_{i=0}^{n} \Sigma^i)|}{|(L_1 \cup L_2) \cap (\bigcup_{i=0}^{n} \Sigma^i)|}$. Further, it gives a closed form for calculating that distance between two unary regular languages.

Besides the stronger results, our work differs from that of [2, 11, 16] in the analysis of the distance functions presented: in particular, one can conclude (as a consequence of Theorem 17) that the above equation is mostly trivial – it returns 0 or 1 "most" of the time.

More recently, Cui *et al* directly address distances between regular languages using a generalization of Jaccard distance [9]. That paper usefully expands the concept of Jaccard distance to regular languages by (1) using entropy to handle infinite sized regular languages (they use the upper limit notion of entropy described above), and (2) allowing operations other than intersection to be used in the numerator. Further, Cui *et al* suggest and prove properties of several specific distance functions between regular languages. The distance functions in this paper do not generalize the Jaccard distance in the same way, but are proven to be metrics or pseudo-metrics.

Ceccherini-Silberstein *et al* investigate the entropy of specific kinds of subsets of regular languages [3]. They present a novel proof of a known fact from Symbolic Dynamics. They use the same upper limit notion of entropy as above. Other entropy formulations include the number of prefixes of a regular language [5], but this has only been proven equivalent to entropy under restricted circumstances.

Symbolic dynamics [19] studies, among other things, an object called a sofic shift. Sofic shifts are analogous to deterministic finite automata and their shift spaces are related to regular languages. The formulation of entropy used in this field does not suffer from issues of potential non-existence. This paper includes a proof that the *topological entropy* of a sofic shift is equivalent to language-centric formulations in this paper: see Theorem 13.

Other related results from symbolic dynamics include an investigation into the computability of a sofic shift's entropy [28] and a discussion of the lack of relationship between entropy and complexity [18]. There is another proposal for the topological entropy of formal languages [26] that is zero for all regular languages (and hence not helpful as a distance function for regular languages).

A probabilistic automaton is an automaton with a probability distribution applied to outgoing transitions from each state. The words of a regular language thus inherit a probability. Using standard distance functions on probability distributions (such as L_p and Kullback-Leibler divergence), several distance functions [7, 8, 21] have been created for probabilistic languages. Note that in this model, the probability of a word exponentially decreases with its length, and hence these distance functions can be effectively estimated by words of bounded length. Chan [4] also describes several distance functions using only words of bounded length. Our paper will uncover features of several distance functions, which will fit nicely into the above frameworks.

$J'_n(L_1, L_2)$	n Jaccard Distance	$\frac{\lvert W_n(L_1 \triangle L_2)\rvert}{\lvert W_n(L_1 \cup L_2)\rvert}$
$J_n(L_1, L_2)$	$n_\leq$ Jaccard Distance	$\frac{\lvert W_{\leq n}(L_1 \triangle L_2)\rvert}{\lvert W_{\leq n}(L_1 \cup L_2)\rvert}$
$J_C(L_1, L_2)$	Cesàro Jaccard	$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} J_i(L_1, L_2)$
$H(L_1, L_2)$	Entropy Distance	$\frac{h(L_1 \triangle L_2)}{h(L_1 \cup L_2)}$
$H_S(L_1, L_2)$	Entropy Sum Distance	$h(L_1 \cap \overline{L_2}) + h(\overline{L_1} \cap L_2)$

Table 1 The distance functions considered in this paper are listed in this table.

2.2 Definitions and Notation

In this paper Σ will denote a set of *symbols* or the alphabet. *Strings* are concatenations of these symbols. All log operations in this paper will be taken base 2. Raising a string to the power n will represent the string resulting from n concatenations of the original string. A similar notion applies to sets. In this notation, Σ^5 represents all strings of length 5 composed of symbols from Σ. The Kleene star, $*$, when applied to a string (or a set) will represent the set containing strings resulting from any number of concatenations of that string (or of strings in that set), including the empty concatenation. Thus, Σ^* represents all possible strings comprised of symbols in Σ, including the empty string.

A *regular language* is a set $L \subset \Sigma^*$ which can be represented by a *Deterministic Finite Automaton*, DFA for short. A *DFA* is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where Q is a set of states, Σ is the set of symbols, δ is a partial function from $Q \times \Sigma$ to Q, $q_0 \in Q$ is the *initial state* and $F \subset Q$ is a set of *final states*. A regular language can also be constructed by recursive applications of concatenation (denoted by placing regular expressions adjacent to one another), disjunction (denoted $|$), and Kleene star (denoted $*$), to strings and the empty string. That this construction and the DFA are equivalent is well known [14].

The DFA $(Q, \Sigma, \delta, q_0, F)$ can be thought of as a directed graph whose vertices are Q with edges from q to q' iff there is an $s \in \Sigma$ such that $q' = \delta(q, s)$. The transition function δ provides a labeling of the graph, where each edge (q, q') is labeled by the symbol s such that $\delta(q, s) = q'$. Note that there may be multiple edges between nodes, each with a different label. The adjacency matrix A for a DFA is the adjacency matrix for the corresponding graph. Thus, entries in A are given by $a_{q,q'}$, where $a_{q,q'}$ is the number of edges from vertex q to vertex q'.

For a regular language L, let $W_n(L)$ denote the set of words in L of length exactly n, i.e. $W_n(L) = L \cap \Sigma^n$, and let $W_{\leq n}(L)$ denote the set of words in L of length at most n, i.e. $W_{\leq n}(L) = L \cap (\bigcup_{i=0}^{n} \Sigma^i)$.

Finally, we will discuss when certain distance functions are metrics. A *metric* on the space X is a function $d : X \times X \to \mathbb{R}$ that satisfies

1. $d(x, y) \geq 0$ with equality if and only if $x = y$ for all $x, y \in X$
2. $d(x, y) = d(y, x)$ for all $x, y \in X$
3. $d(x, z) \leq d(x, y) + d(y, z)$ for all $x, y, z \in X$.

An *ultra-metric* is a stronger version of a metric, with the triangle inequality (the third condition above) replaced with the ultra-metric inequality: $d(x, z) \leq \max\{d(x, y), d(y, z)\}$ for all $x, y, z \in X$. Also, there exists a weaker version, called a *pseudo-metric*, which allows $d(x, y) = 0$ when $x \neq y$.

3 Jaccard Distances

The Jaccard distance is a well-known distance function between finite sets. For finite sets A and B, the *Jaccard distance* between them is given by $\frac{|A \triangle B|}{|A \cup B|} = 1 - \frac{|A \cap B|}{|A \cup B|}$ where $A \triangle B$ represents the symmetric difference between the two sets (if $A \cup B = \emptyset$ then the Jaccard distance is 0). This classical Jaccard distance is not defined for infinite sets and as such, is not a suitable distance function for infinite regular languages and will need to be modified.

3.1 Jaccard Distances using W_n and $W_{\leq n}$

A natural method for applying Jaccard distance to regular languages is to fix n, defined as follows:

▶ **Definition 1** (n Jaccard Distance)**.** Suppose L_1 and L_2 are regular languages. Define the n *Jaccard distance* by $J'_n(L_1, L_2) = \frac{|W_n(L_1 \triangle L_2)|}{|W_n(L_1 \cup L_2)|}$ if $|W_n(L_1 \cup L_2)| > 0$, otherwise $J'_n(L_1, L_2) = 0$.

For fixed n, the above is a pseudo-metric since it is simply the Jaccard distance among sets containing only length n strings. The following proposition points out one deficiency of J'_n.

▶ **Proposition 2.** There exists a set $S = \{L_1, L_2, L_3\}$ of infinite unary regular languages with $L_2, L_3 \subset L_1$ such that for all n there exists an $i \neq j$ such that $J'_n(L_i, L_j) = 0$.

One may also use $W_{\leq n}$ in the definition of a Jaccard-based distance function.

▶ **Definition 3** ($n_{\leq}$ Jaccard Distance)**.** For regular languages L_1 and L_2, define the $n_{\leq}$ *Jaccard distance* by $J_n(L_1, L_2) = \frac{\left|W_{\leq n}(L_1 \triangle L_2)\right|}{\left|W_{\leq n}(L_1 \cup L_2)\right|}$ if $|W_{\leq n}(L_1 \cup L_2)| > 0$, otherwise $J_n(L_1, L_2) = 0$.

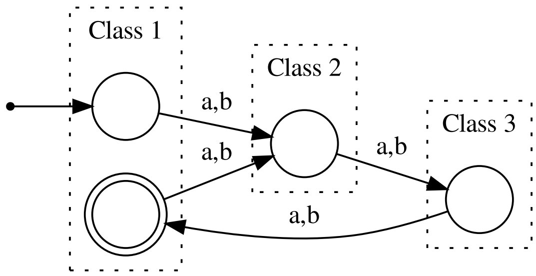
The issue with J'_n pointed out by Proposition 2 can be proven to not be a problem for J_n: see the first point of Theorem 4. On the other hand, the second point of Theorem 4 shows that no universal n exists.

▶ **Theorem 4.** *The function J_n defined above is a pseudo-metric and satisfies the following:*
1. *Let $S = \{L_1, \ldots, L_k\}$ be a set of regular languages. There exists an n such that J_n is a metric over S. Moreover, we may choose n such that $n \leq \max_{i,j}(s(L_i)+1)(s(L_j)+1) - 1$ where $s(L_i)$ represents the number of states in the minimal DFA corresponding to L_i.*
2. *For any fixed n there exist regular languages L, L' with $L \neq L'$ such that $J_n(L, L') = 0$.*

For any pseudo-metric, the relation $d(x, y) = 0$ is an equivalence relation. Thus, if we mod out by this equivalence relation, the pseudo-metric becomes a metric.

Due to the fact that one must choose a fixed n, J_n and J'_n cannot account for the infinite nature of regular languages. Limits based on J_n and J'_n are a natural next step. However, the natural limits involving J'_n and J_n do not always exist. An example showing this was given for J'_n in the beginning of the introduction (Section 1). A similar example applies to J_n. Consider the languages given by $L_1 = (a|b)^*$ and $L_2 = ((a|b)^2)^*$ $(\Sigma = \{a, b\})$. For these languages, $\lim_{n\to\infty} J_{2n}(L_1, L_2) = 2/3$ and $\lim_{n\to\infty} J_{2n+1}(L_1, L_2) = 1/3$. Hence, $\lim_{n\to\infty} J_n(L_1, L_2)$ does not exist.

The next theorem gives conditions for when the limit of J'_n exists as n goes to infinity. Before the theorem is stated we will need some more terminology. Suppose L is a regular language and M is the corresponding DFA. This DFA is a labeled directed graph. An *irreducible component* of M is a strongly connected component of the graph. That is, an irreducible component is composed of a maximal set of vertices such that for any pair, there is a directed path between them.



$$A^i = \begin{cases} \begin{pmatrix} 0 & 2^i & 0 & 0 \\ 0 & 0 & 2^i & 0 \\ 0 & 0 & 0 & 2^i \\ 0 & 2^i & 0 & 0 \end{pmatrix} & \text{if } i \equiv 1 \bmod 3 \\ \begin{pmatrix} 0 & 0 & 2^i & 0 \\ 0 & 0 & 0 & 2^i \\ 0 & 2^i & 0 & 0 \\ 0 & 0 & 2^i & 0 \end{pmatrix} & \text{if } i \equiv 2 \bmod 3 \\ \begin{pmatrix} 0 & 0 & 2^i & 0 \\ 0 & 2^i & 0 & 0 \\ 0 & 0 & 2^i & 0 \\ 0 & 0 & 0 & 2^i \end{pmatrix} & \text{if } i \equiv 0 \bmod 3 \end{cases}$$

Figure 1 The DFA for a period 3 language and the associated adjacency matrix raised to the i^{th} power.

The *period* of an irreducible graph (or associated adjacency matrix) is the largest integer p such that the vertices can be grouped into classes $Q_0, Q_1, \ldots, Q_{p-1}$ such that if $x \in Q_i$, then all of the out neighbors of x are in Q_j, where $j = i + 1(mod\ p)$. The period of a reducible graph is the least common multiple of the periods of its irreducible components. See Figure 1 for an example of a regular language whose DFA has period 3. For a more formal definition of periodicity see [19]. If the graph (or matrix) has period 1 it will be called *aperiodic*. Matrices that are irreducible and aperiodic are called *primitive*. The definition of primitive presented here is equivalent to the condition that there is an n such that all entries of the adjacency matrix A raised to the n-th power (A^n) are positive [20]. This is illustrated in Figure 1, where the graph is periodic and reducible and all powers of that matrix contain multiple zeroes.

▶ **Theorem 5.** *Suppose L_1 and L_2 are regular languages. If each irreducible component of the DFA associated to $L_1 \triangle L_2$ and $L_1 \cup L_2$ are aperiodic, then $\lim_{n\to\infty} J'_n(L_1, L_2)$ converges.*

Let us build intuition prior to proving Theorem 5, which will also frame the question of convergence in the next subsection. We will first discuss Theorem 5 in the case where the DFA associated to regular languages $L_1 \triangle L_2$ and $L_1 \cup L_2$ are primitive. Suppose $A_\triangle$ and $A_\cup$ are the adjacency matrices for $L_1 \triangle L_2$ and $L_1 \cup L_2$ respectively. Perron-Frobenius theory tells us that the eigenvalue of largest modulus of a primitive matrix is real and unique. Let $(v_\triangle, \lambda_\triangle)$ and $(v_\cup, \lambda_\cup)$ be eigenpairs composed of the top eigenvalues for $A_\triangle$ and $A_\cup$ respectively. Notice that $i_\triangle A^n_\triangle f_\triangle$, where $i_\triangle$ is the row vector whose jth entry is 1 if j is an initial state in $A_\triangle$ and 0 otherwise (a similar definition for final states defining column vector $f_\triangle$ holds), represents words in $L_1 \triangle L_2$ of length n. If we write $f_\triangle = c_1 v_\triangle + c_2 w$ and $f_\cup = d_1 v_\cup + d_2 y$, then $i_\triangle A^n_\triangle f_\triangle$ converges to $\lambda^n_\triangle c_1 i_\triangle v_\triangle$, and $i_\cup A^n_\cup f_\cup$ converges to $\lambda^n_\cup d_1 i_\cup v_\cup$ as n goes to infinity. This convergence is guaranteed because $\lambda_\cup$ and $\lambda_\triangle$ are unique top eigenvalues. Thus,

$$\lim_{n\to\infty} J'_n(L_1, L_2) = \lim_{n\to\infty} \left(\frac{\lambda_\triangle}{\lambda_\cup}\right)^n \frac{c_1 i_\triangle v_\triangle}{d_1 i_\cup v_\cup}$$

and the limit converges ($\lambda_\triangle \leq \lambda_\cup$ because $L_1 \triangle L_2 \subseteq L_1 \cup L_2$).

The general case of Theorem 5, which does not assume $L_1 \triangle L_2$ and $L_1 \cup L_2$ have irreducible matrices, is more complicated. However, the outline of the argument is the same, and we will sketch it here. The key difference is the use of newer results. An understanding of the asymptotic behavior of A^n for large n was finally beginning to be developed several decades after Chomsky and Miller investigated regular languages. In 1981 Rothblum [23] proved that for each non-negative matrix A with largest eigenvalue λ, there exists $q \geq 1$ (which happens to be the period of A) and polynomials $S_0(x), S_1(x), \ldots, S_{q-1}(x)$ (whose domain is the set of real numbers and whose coefficients are matrices) such that for all whole numbers $0 \leq k \leq q-1$ we have that $\lim_{n\to\infty} (A/\lambda)^{qn+k} - S_k(n) = 0$. We will refer to this result later in the paper, where we will simply call it **Rothblum's Theorem** (a slow treatment of this theory with examples can be found in [24]). So the rest of the proof to Theorem 5 is observing that $q = 1$ in the case we are interested in, and so $\lim_{n\to\infty} J'_n(L_1, L_2)$ converges.

3.2 Cesàro Jaccard

For a sequence of numbers $a_1, a_2, \ldots$, a Cesàro summation is $\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} a_i$ when the limit exists. The intuition behind a Cesàro summation is that it may give the "average value" of the limit of the sequence, even when the sequence does not converge. For example, the sequence $a_j = e^{\alpha i j}$ (where $i^2 = -1$) has Cesàro summation 0 for all real numbers $\alpha \neq 0$. This follows from the fact that rotations of the circle are uniquely ergodic [13]. Not all sequences have a Cesàro summation, even when we restrict our attention to sequences whose values lie in $[0, 1]$. For example, the sequence b_i, where $b_i = 1$ when $2^{2n} < i < 2^{2n+1}$ for some $n \in \mathbb{N}$ and $b_i = 0$ otherwise has no Cesàro summation. However, we will be able to show that the Cesàro average of Jaccard distances does exist.

To that end, another limit based distance is the Cesàro average of the J_n or J'_n.

▶ **Definition 6** (Cesàro Jaccard Distance). Suppose L_1 and L_2 are regular languages. Define the *Cesàro Jaccard distance* by $J_C(L_1, L_2) = \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} J_i(L_1, L_2)$.

The Cesàro Jaccard distance is theoretically better than the above suggestions in Section 3.1 since it can be shown to exist for all regular languages.

▶ **Theorem 7.** *Let L_1 and L_2 be two regular languages. Then, $J_C(L_1, L_2)$ is well-defined. That is,*
$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} J_i(L_1, L_2)$ exists.

We will breifly sketch the proof to Theorem 7. Recall that $|W_n(L_1 \triangle L_2)|$ and $|W_n(L_1 \cup L_2)|$ can be calculated using powers of specific matrices. If we take Q to be the least common multiple of the period from each of the matrices associated with $|W_n(L_1 \triangle L_2)|$ and $|W_n(L_1 \cup L_2)|$, we can immediately see that $\lim_{n\to\infty} J'_{Qn+k}(L_1, L_2)$ exists, via Rothblum's Theorem. Moreover, it will equal zero if they have different values for the largest eigenvalue or the degree of $S_k(x)$. But if they have the same value for the largest eigenvalue and degree of $S_k(x)$, then $\lim_{n\to\infty} J'_{Qn+k}(L_1, L_2)$ will be the ratio of the leading coefficients of the polynomials $S_k(x)$ for the two matrices. The proof finishes by observing that $J'_C(L_1, L_2) = \frac{1}{n}\sum_{i=1}^{n} J'_i(L_1, L_2)$ will be the average of these values.

We will require a new result to show that the more interesting value $J_C(L_1, L_2)$ is well-defined (part (2) of the theorem is similar to a result in [23]).

▶ **Theorem 8.** *Let A be the adjacency matrix for a DFA representing a regular language L, and let λ be the largest eigenvalue of A. Let q and $S_0(x), S_1(x), \ldots, S_{q-1}(x)$ be as in Rothblum's theorem; let d be the largest degree of the polynomials $S_0(x), S_1(x), \ldots, S_{q-1}(x)$. Let $s_\ell = \lim_{n\to\infty} n^{-(d+1)}\sum_{i=1}^{n} S_\ell(i)$ and $t_\ell = \lim_{n\to\infty} n^{-d} S_\ell(n)$.*

1. *If $\lambda < 1$, then L is finite.*

2. *If* $\lambda = 1$*, then* $\lim_{n\to\infty} \frac{1}{n^{d+1}} \sum_{i=1}^{n} A^i = \sum_{i=0}^{q-1} s_\ell$.
3. *If* $\lambda > 1$*, then* $\lim_{n\to\infty} \frac{1}{(qn+k)^d} \lambda^{-(qn+k)} \sum_{i=1}^{qn+k} A^i = \frac{1}{1-\lambda^{-q}} \sum_{\ell=k-q+1}^{k} \lambda^{\ell-k} t_\ell$ *where the indices of the* t_i *are taken modulo* q*.*

Using our new result in place of Rothblum's theorem, we now see that $J_C(L_1, L_2)$ is well-defined. Note that in $J'_C(L_1, L_2)$ each congruence class k is handled independently and the final answer is the average of such results. On the other hand, in $J_C(L_1, L_2)$ each congruence class k has a limit that is a combination of results from all of the congruence classes. Thus the total answer is dominated by the overall asymptotic behavior and not just small periodic undercurrents. We illustrate this point via the next example.

▶ **Example 9.** Let $L_1 = ((a|b)^2)^*|c^*$ and $L_2 = ((a|b)^2)^*|d^*$. The languages L_1 and L_2 have $((a|b)^2)^*$ in common and so mutually shared words up to length n grow exponentially. The languages disagree on c^* and d^*, whose words only grow polynomially. Hence, L_1 and L_2 are very similar and should have a small distance. However, J'_C gives equal weight to words of even length and odd length, even though the languages are mostly made up of even-length words.

Rigorously, we have that $\lim_{n\to\infty} J_{2n}(L_1, L_2) = 0$ and $\lim_{n\to\infty} J'_{2n}(L_1, L_2) = 0$. Furthermore, $\lim_{n\to\infty} J_{2n+1}(L_1, L_2) = 0$ and $\lim_{n\to\infty} J'_{2n+1}(L_1, L_2) = 1$. Thus, $J_C(L_1, L_2) = 0$, while $J'_C(L_1, L_2) = \frac{1}{2}$.

We conclude this section with a fact about the Cesáro Jaccard distance.

▶ **Fact 10.** The Cesàro Jaccard distance inherits the pseudo-metric property from J_n.

4 Entropy

In this section we develop the idea of topological entropy for a certain type of dynamical system and show how it relates to a quantity that we have identified as the language entropy. Then, we will show how Cesáro Jaccard is related to entropy.

4.1 Topological Entropy

Topological entropy is a concept from dynamical systems where the space is a compact metric space and the map defined there is continuous [19]. In dynamics, successive applications of the map are applied and the long term behavior of the system is studied. An orbit of a point x for the map T is the set $\{T^n(x) : n \in \mathbb{Z}\}$. Topological entropy is an abstract concept meant to determine the exponential growth of distinguishable orbits of the dynamical system up to arbitrary scale. A positive quantity for topological entropy reflects chaos in the system [1]. This concept was motivated by Kolmogorov and Sinai's theory of measure-theoretic entropy in ergodic theory [15, 29], which in turn is related to Shannon entropy [27]. An example of a topological dynamical system is a sofic shift, which is a symbolic system that is intricately related to DFA. Instead of defining the topological entropy of a sofic shift symbolically, which is classical, we will use the graph theoretic description.

A *sofic shift* can be thought of as the space of biinfinite walks (i.e. walks with no beginning and no end) on a right-solving labeled directed graph (a right-solving labeled graph has a unique label for each edge leaving a given node). Suppose G is a directed graph where V is the set of vertices and E is the set of edges of G. Furthermore, suppose that every edge in E is labeled with a symbol from Σ, and that there is at most one outgoing edge from each vertex with a given label (i.e. *right-solving*). Note that this construction is similar to a DFA,

aaa, *aaba*, *ba*, *aaaaaaa*, *aaabaabaa*, ... *aabaabaabaaaabaaaaaabaaaaab* ...

Figure 2 A DFA with some accepted strings and a sofic shift with a portion of a derived biinfinite string.

however there are no initial and final states. A biinfinite walk on G with a specified base vertex is an infinite walk in both directions (forward and backward) from the base vertex on the graph. This biinfinite walk corresponds to a biinfinite string of symbols from Σ. See Figure 2.

We will call a finite block of symbols *admissible* if there is a biinfinite string of symbols corresponding to a biinfinite walk on G and this finite block appears somewhere within the biinfinite string. Note that all sufficiently long words in the DFA's language will contain a substring of almost the same length that is an admissible block, while not all admissible blocks will be in the associated DFA's language. Denote the set of admissible blocks of length n corresponding to G by $B_n(G)$. The *topological entropy* of the sofic shift represented by the right-solving labeled graph G is denoted by $h_t(G)$ and is defined by

$$h_t(G) = \lim_{n\to\infty} \frac{\log |B_n(G)|}{n}.$$

Using Perron-Frobenius theory it has been proven that the topological entropy of a sofic shift represented by a right-solving labeled graph G is equal to the log base 2 of the spectral radius of the adjacency matrix of G [19]. That is, the topological entropy is given by the log of the adjacency matrix's largest modulus eigenvalue. Algorithms for computing eigenvalues are well known and run in time polynomial in the width of the matrix [12].

As you can see, sofic shifts are very similar to DFA. Given a DFA, M, one can construct a sofic shift by thinking of M as a labeled directed graph and creating the *trim graph* by removing all states that are not part of an accepting path. Information regarding initial and final states is no longer needed. Note that the graph M is naturally right-solving because of the determinism of DFA. It is also easiest to remove from M all vertices that do not have both an outgoing and incoming edge (since we are now interested in biinfinite walks). The resulting graph is called the *essential graph*. At this point one is free to apply the above definition and compute the topological entropy of the sofic shift corresponding to the DFA. This quantity can be computed by analyzing the irreducible components.

▶ **Theorem 11** ([19]). *Suppose that G is the labeled directed graph associated to a sofic shift. If $G_1, \ldots, G_k$ are the irreducible components of G, then $h_t(G) = \max_{1\le i\le k} h_t(G_i)$.*

In the next subsection we will introduce the language entropy and show that it is the same as the topological entropy of the sofic shift corresponding to a DFA.

4.2 Language Entropy

Traditionally, the entropy of a regular language L (also called the *channel capacity* [6] or *information rate* [10]) is defined as $\limsup_{n\to\infty} \frac{\log|W_n(L)|}{n}$. This limit may not exist and so an upper limit is necessary. We will show that this upper limit is realized by the topological entropy of the corresponding sofic shift and define another notion of language entropy, which is preferable since an upper limit is not necessary.

▶ **Definition 12** (Language Entropy)**.** Given a regular language L define the *language entropy* by $h(L) = \lim_{n\to\infty} \frac{\log|W_{\leq n}(L)|}{n}$.

▶ **Theorem 13.** *Let L be a non-empty regular language over the set of symbols Σ, and let G be the labeled directed graph of the associated sofic shift. We have that*

$$\limsup_{n\to\infty} \frac{\log|W_n(L)|}{n} = h_t(G).$$

Moreover, for a fixed language L there exists a constant c such that there is an increasing sequence of integers n_i satisfying $0 < n_{i+1} - n_i \leq c$ and

$$\lim_{i\to\infty} \frac{\log|W_{n_i}(L)|}{n_i} = h_t(G).$$

As a corollary to this theorem we obtain an important statement regarding the connection between topological entropy (from dynamical systems) and language entropy (similar to Shannon's channel capacity). The following statement is consistent with remarks made by Chomsky and Miller [6] that involved undefined assumptions; we show rigorously that this formula is correct for all DFA.

▶ **Corollary 14.** *Let L be a non-empty regular language over the set of symbols Σ, and let G be the labeled directed graph of the associated sofic shift. Then,*

$$h(L) = \lim_{n\to\infty} \frac{\log|W_{\leq n}(L)|}{n} = h_t(G).$$

There are some simple properties of language entropy which will be useful later. The first is a simple re-phrasing of Corollary 14.

▶ **Lemma 15.** *For any regular language L, we have that $|W_{\leq n}(L)| = 2^{n(h(L)+o(1))}$.*

▶ **Lemma 16.** *Suppose L_1 and L_2 are regular languages over Σ. The following hold:*
1. *If $L_1 \subseteq L_2$, then $h(L_1) \leq h(L_2)$.*
2. $h(L_1 \cup L_2) = max(h(L_1), h(L_2))$
3. $\max(h(L_1), h(\overline{L_1})) = \log|\Sigma|$
4. *If $h(L_1) < h(L_2)$, then $h(L_2 \setminus L_1) = h(L_2)$.*
5. *If L_1 is finite, then $h(L_1) = 0$.*

4.3 Relationship between Entropy and Cesáro Jaccard

In Section 3.2 we proved that the Cesàro Jaccard distance is well-defined. As you will see, Cesáro Jaccard and entropy are mostly disjoint in what they measure.

▶ **Theorem 17.** *Let L_1, L_2 be two regular languages.*
1. *If $h(L_1 \triangle L_2) \neq h(L_1 \cup L_2)$, then $J_C(L_1, L_2) = 0$.*
2. *If $h(L_1 \cap L_2) \neq h(L_1 \cup L_2)$, then $J_C(L_1, L_2) = 1$.*

3. *If* $0 < J_C(L_1, L_2) < 1$*, then the following equal each other:* $h(L_1),\ h(L_2),\ h(L_1 \cap L_2),\ h(L_1 \triangle L_2),\ h(L_1 \cup L_2)$.

To better understand this theorem, consider the following examples corresponding to the three cases of the theorem: (1) let $L_1 = ((a|b)^2)^*|c^*$ and $L_2 = ((a|b)^2)^*|d^*$ as in Example 9, (2) let $L_1 = (a|b)^*|c^*$ and $L_2 = (d|e)^*|c^*$, and (3) let $L_1 = (aa)^*$ and $L_2 = a^*$ as in the Introduction.

5 Entropy Distances

Entropy provides a natural method for dealing with the infinite nature of regular languages. Because it is related to the eigenvalues of the regular language's DFA, it is computable in polynomial time given a DFA for the language. Note that the DFA does not have to be minimal. We can therefore compute the entropy of set-theoretic combinations of regular languages (intersection, disjoint union, etc) and use those values to determine a distance between the languages.

5.1 Entropy Distance

A natural Jaccard-esque distance function based on entropy is the entropy distance.

▶ **Definition 18** (Entropy Distance)**.** Suppose L_1 and L_2 are regular languages. Define the *entropy distance* to be $H(L_1, L_2) = \frac{h(L_1 \triangle L_2)}{h(L_1 \cup L_2)}$ if $h(L_1 \cup L_2) > 0$, otherwise $H(L_1, L_2) = 0$.

This turns out to be equivalent to a Jaccard limit with added log operations:

▶ **Corollary 19.** *Suppose L_1 and L_2 are regular languages. The following relation holds:*

$$\lim_{n\to\infty} \frac{\log |W_{\leq n}(L_1 \triangle L_2)|}{\log |W_{\leq n}(L_1 \cup L_2)|} = H(L_1, L_2).$$

Note that H is not always a good candidate for a distance function as it only produces non-trivial results for languages that have the same entropy.

▶ **Proposition 20.** Suppose L_1 and L_2 are regular languages. If $h(L_1) \neq h(L_2)$, then $H(L_1, L_2) = 1$.

As further evidence that H is not a good candidate for a distance function, we show it is an ultra-pseudo-metric. The ultra-metric condition, i.e. $d(x, z) \leq \max(d(x, y), d(y, z))$, is so strong that it can make it difficult for the differences encoded in the metric to be meaningful for practical applications.

▶ **Theorem 21.** *The function H is an ultra-pseudo-metric.*

5.2 Entropy Sum

In this subsection we will define a new (and natural) distance function for infinite regular languages. We call this distance function the *entropy sum distance*. We will prove that not only is this distance function a pseudo-metric, it is also sometimes granular. Granularity lends insight into the quality of a metric. Intuitively, granularity means that for any two points in the space, you can find a point between them. A metric d on the space X is *granular* if for every two points $x, z \in X$, there exists $y \in X$ such that $d(x, y) < d(x, z)$ and $d(y, z) < d(x, z)$, i.e. $d(x, z) > \max(d(x, y), d(y, z))$.

▶ **Definition 22** (Entropy Sum Distance)**.** Suppose L_1 and L_2 are regular languages. Define the *entropy sum distance* to be $H_S(L_1, L_2) = h(L_1 \cap \overline{L_2}) + h(\overline{L_1} \cap L_2)$.

The entropy sum distance was inspired by first considering the entropy of the symmetric difference directly, i.e. $h(L_1 \triangle L_2)$. However, since entropy measures the entropy of the most complex component (Theorem 11), more information is gathered by using a sum as above in the definition of entropy sum.

▶ **Theorem 23.** *The function H_S is a pseudo-metric.*

The next two propositions display when granularity is achieved and when it is not.

▶ **Proposition 24.** Let L_1 and L_2 be regular languages such that $h(L_1 \cap \overline{L_2}), h(\overline{L_1} \cap L_2) > 0$. Then, there exists two regular languages $R_1 \neq R_2$ such that $H_S(L_1, L_2) > \max(H_S(L_1, R_i), H_S(R_i, L_2))$ for each i.

▶ **Proposition 25.** Let L_1 and L_2 be regular languages such that $h(\overline{L_1} \cap L_2) = 0$. For all regular languages L we have that $H_S(L_1, L_2) \leq \max(H_S(L_1, L), H_S(L, L_2))$.

6 Conclusion and Future Work

This paper has covered some issues related to the entropy of regular languages and the distance between regular languages. It has proven correct the common upper limit formulation of language entropy and has provided a limit based entropy formula that can be shown to exist. Jaccard distance was shown to be related to language entropy, and various limit based extensions of the Jaccard distance were shown to exist or not exist. The natural entropy based distance function was shown to be an ultra-pseudo-metric, and some facts were proven about the function that show it likely to be impractical. Finally, the paper introduces an entropy-based distance function and proves that function to be a pseudo-metric, as well as granular under certain conditions.

In this paper several formulations of entropy are developed, and it is natural to consider which would be the best to use. In a practical sense it does not matter since all formulations are equivalent (Theorem 13) and can be computed using Shannon's determinant-based method [27]. However, conceptually, it can be argued that $\lim_{n\to\infty} \frac{log|W_{\leq n}(L)|}{n}$ is the preferable formulation. First, there is a notational argument that prefers using limits that exist. This is a limit that exists (Corollary 14), whereas many other limit formulations do not. Second, this limit captures more readily the concept of "number of bits per symbol" that Shannon intended. Because regular languages can have strings with staggered lengths, using W_n forces the consideration of possibly empty sets of strings of a given length. This creates dissonance when the language has non-zero entropy. Instead, the monotonically growing $W_{\leq n}$ more clearly encodes the intuition that the formulation is expressing the number of bits needed to express the next symbol among all words in the language.

Apart from expanding to consider context-free languages and other languages ([10]), one investigation that is absent from this paper is the determination of similarity between languages that are disjoint but obviously similar (i.e. aa^* and ba^*). A framework for addressing such problems is provided in [9], but finding metrics capturing such similarities can be fodder for future efforts.

References

1 F. Blanchard, E. Glasner, S. Kolyada, and A. Maass. On Li-Yorke pairs. *J. Reine Angew. Math.*, 547:51 – 68, 2002.

2 M. Bodirsky, T. Gärtner, T. von Oertzen, and J. Schwinghammer. Efficiently computing the density of regular languages. In *LATIN 2004: Theoretical informatics*, volume 2976 of *Lecture Notes in Comput. Sci.*, pages 262–270. Springer, Berlin, 2004. URL: `http://dx.doi.org/10.1007/978-3-540-24698-5_30`, `doi:10.1007/978-3-540-24698-5_30`.

3 T. Ceccherini-Silberstein, A. Machì, and F. Scarabotti. On the entropy of regular languages. *Theoretical computer science*, 307(1):93 – 102, 2003.

4 C. Chan, M. Garofalakis, and R. Rastogi. Re-tree: an efficient index structure for regular expressions. *The VLDB Journal—The International Journal on Very Large Data Bases*, 12(2):102 – 119, 2003.

5 C. Chang. Algorithm for the complexity of finite automata. *31st Workshop on Combinatorial Mathematics and Computation Theory*, pages 216 – 220, 2014.

6 N. Chomsky and G. Miller. Finite state languages. *Information and Control*, 1(2):91 – 112, 1958. URL: `http://www.sciencedirect.com/science/article/pii/S0019995858900822`, `doi:http://dx.doi.org/10.1016/S0019-9958(58)90082-2`.

7 C. Cortes, M. Mohri, and A. Rastogi. On the computation of some standard distances between probabilistic automata. In *Implementation and application of automata*, volume 4094 of *Lecture Notes in Comput. Sci.*, pages 137–149. Springer, Berlin, 2006. URL: `http://dx.doi.org/10.1007/11812128_14`, `doi:10.1007/11812128_14`.

8 C. Cortes, M. Mohri, A. Rastogi, and M. Riley. Efficient computation of the relative entropy of probabilistic automata. In *LATIN 2006: Theoretical informatics*, volume 3887 of *Lecture Notes in Comput. Sci.*, pages 323–336. Springer, Berlin, 2006. URL: `http://dx.doi.org/10.1007/11682462_32`, `doi:10.1007/11682462_32`.

9 C. Cui, Z. Dang, T. Fischer, and O. Ibarra. Similarity in languages and programs. *Theoretical Computer Science*, 498:58 – 75, 2013.

10 C. Cui, Z. Dang, T. Fischer, and O. Ibarra. Information rate of some classes of non-regular languages: an automata-theoretic approach (extended abstract). In *Mathematical foundations of computer science 2014. Part I*, volume 86343 of *Lecture notes in Comput. Sci.*, pages 232 – 243. Springer, Heidelberg, 2014.

11 J. Dassow, G. Martín Reyes, and F. Vico Vela. A similarity measure for cyclic unary regular languages. 2009.

12 J. Francis. The QR transformation a unitary analogue to the LR transformation — part 1. *The Computer Journal*, 4(3):265 – 271, 1961.

13 B. Hasselblatt and A. Katok. *A first course in dynamics: With a panorama of recent developments*. Cambridge University Press, New York, 2003.

14 J. Hopcroft and J. Ullman. *Introduction to automata theory, languages, and computation*. Addison-Wesley Publishing Company, Inc., 1979.

15 A. Kolmogorov. Entropy per unit time as a metric invariant of automorphisms. In *Dokl. Akad. Nauk SSSR*, volume 124, pages 754 – 755, 1959.

16 J. Kozik. Conditional densities of regular languages. In *Proceedings of the Second Workshop on Computational Logic and Applications (CLA 2004)*, volume 140 of *Electron. Notes Theor. Comput. Sci.*, pages 67–79 (electronic). Elsevier, Amsterdam, 2005. URL: `http://dx.doi.org/10.1016/j.entcs.2005.06.023`, `doi:10.1016/j.entcs.2005.06.023`.

17 W. Kuich. On the entropy of context-free languages. *Information and Control*, 16(2):173 – 200, 1970.

18 W. Li. On the relationship between complexity and entropy for Markov chains and regular languages. *Complex systems*, 5(4):381 – 399, 1991.

19 D. Lind and B. Marcus. *Symbolic dynamics and coding*. Cambridge, 1995.

20 J. Marklof and C. Ulcigrai. Lecture notes for dynamical systems and ergodic theory, 2015-2016. http://www.maths.bris.ac.uk/~majm/DSET/index.html.

21 M-J. Nederhof and G. Satta. Computation of distances for regular and context-free probabilistic languages. *Theoret. Comput. Sci.*, 395(2-3):235–254, 2008. URL: `http://dx.doi.org/10.1016/j.tcs.2008.01.010`, `doi:10.1016/j.tcs.2008.01.010`.

22 A. Parker, K. Yancey, and M. Yancey. Regular language distance and entropy. *arXiv*, 602.07715, 2015.

23 U. Rothblum. Expansion of sums of matrix powers. *SIAM Review*, 23:143 – 164, 1981.

24 U. Rothblum. Chapter 9, nonnegative matrices and stochastic matrices. In *Handbook of Linear Algebra.* (eds: L. Hogben), Chapman and Hall / CRC, 2007.

25 A. Salomaa and M. Soittola. Automata-theoretic aspects of formal power series. Springer-Verlag, 1978.

26 F. Schneider and D. Borchmann. Topological entropy of formal languages. *arXiv*, 1507.03393, 2015.

27 C. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379 – 423, 1948.

28 J. Simonsen. On the computability of the topological entropy of subshifts. *Discrete mathematics and theoretical computer science*, 8(1):83 – 95, 2006.

29 Y. Sinai. On the notion of entropy of a dynamical system. In *Dokl Akad Nauk SSSR*, volume 124, pages 768 – 771, 1959.