Combinatorics on Words: Pattern Avoidance

Nikhil Ravishankar

Euler Circle

July 10, 2025

Overview

Applications

Combinatorics on words has some applications to other sciences, including:

Applications

Combinatorics on words has some applications to other sciences, including:

- Theoretical CS (string processing, data compression, error detection)

Applications

Combinatorics on words has some applications to other sciences, including:

- Theoretical CS (string processing, data compression, error detection)
- Bioinformatics (analyzing DNA sequences with words)

Applications

Combinatorics on words has some applications to other sciences, including:

- Theoretical CS (string processing, data compression, error detection)
- Bioinformatics (analyzing DNA sequences with words)
- Physics (encoding scenarios in symbolic dynamics)

Preliminaries

Definitions

Definition

An *alphabet* is a finite set of symbols (called *letters*), often denoted Σ.
Examples include the standard English alphabet $\{a, b, c, \ldots z\}$ and the set $\{0, 1\}$.

Definitions

Definition

An *alphabet* is a finite set of symbols (called *letters*), often denoted Σ.
Examples include the standard English alphabet $\{a, b, c, \ldots z\}$ and the set $\{0, 1\}$.

Definition

A *word* is a sequence of letters, all of which are elements of Σ.
Words may be finite, infinite in one direction, or infinite in both directions. For a finite word w, $|w|$ is its length. We denote by Σ^* the set of all finite words that can be formed from the letters in Σ.

Definitions

Definition

An *alphabet* is a finite set of symbols (called *letters*), often denoted Σ.
Examples include the standard English alphabet $\{a, b, c, \ldots z\}$ and the set $\{0, 1\}$.

Definition

A *word* is a sequence of letters, all of which are elements of Σ.
Words may be finite, infinite in one direction, or infinite in both directions. For a finite word w, $|w|$ is its length. We denote by Σ^* the set of all finite words that can be formed from the letters in Σ.

Remark

There is a unique *empty word*, denoted ε, which has length 0.

More Definitions

Definition

A *factor* of a word w is any contiguous subword of w. A *prefix* is a factor found at the beginning of w and a *suffix* is a factor found at the end of w.

More Definitions

Definition

A *factor* of a word w is any contiguous subword of w. A *prefix* is a factor found at the beginning of w and a *suffix* is a factor found at the end of w.

Example

The word *bababba* has the factor *b* as a prefix,

bababba

More Definitions

Definition

A *factor* of a word w is any contiguous subword of w. A *prefix* is a factor found at the beginning of w and a *suffix* is a factor found at the end of w.

Example

The word *bababba* has the factor *b* as a prefix, *bba* as a suffix,

bababba

More Definitions

Definition

A *factor* of a word w is any contiguous subword of w. A *prefix* is a factor found at the beginning of w and a *suffix* is a factor found at the end of w.

Example

The word *bababba* has the factor *b* as a prefix, *bba* as a suffix, *ba* as a prefix and a suffix,

bababba

More Definitions

Definition

A *factor* of a word w is any contiguous subword of w. A *prefix* is a factor found at the beginning of w and a *suffix* is a factor found at the end of w.

Example

The word *bababba* has the factor *b* as a prefix, *bba* as a suffix, *ba* as a prefix and a suffix, and does not have the word *aa* as a factor.

bababba
(aa doesn't appear)

Final Definitions

Factors can be "multiplied" via concatenation. Concatenating u with v is uv. Raising a word to a power is repeated concatenation.

Final Definitions

Factors can be "multiplied" via concatenation. Concatenating u with v is uv. Raising a word to a power is repeated concatenation.

Example

- $u = ab, v = ba \longrightarrow uv = abba$
- $u = abba, v = \varepsilon \longrightarrow uv = abba$
- $u = ab \longrightarrow u^3 = ababab$

Final Definitions

Factors can be "multiplied" via concatenation. Concatenating u with v is uv. Raising a word to a power is repeated concatenation.

Example

- $u = ab, v = ba \longrightarrow uv = abba$
- $u = abba, v = \varepsilon \longrightarrow uv = abba$
- $u = ab \longrightarrow u^3 = ababab$

Definition

A *morphism* is a function $h : \Sigma^* \to \Gamma^*$ that is always distributive over concatenation. In other words, for all words $u, v \in \Sigma^*$, $h(uv) = h(u)h(v)$.

Patterns

Patterns

Definition

Consider a second alphabet Δ. The letters in this alphabet are called *variables*, and words in Δ^* are called *patterns.* A finite word $w \in \Sigma^*$ follows a pattern p if there is a way to create w by substituting finite non-empty words in Σ^* for each variable of p.

Patterns

Definition

Consider a second alphabet Δ. The letters in this alphabet are called *variables*, and words in Δ^* are called *patterns.* A finite word $w \in \Sigma^*$ follows a pattern p if there is a way to create w by substituting finite non-empty words in Σ^* for each variable of p.

Example

The word *abab* follows the patterns xx, xy, and $xyzy$, but not xxx.

Patterns

Definition

Consider a second alphabet Δ. The letters in this alphabet are called *variables*, and words in Δ^* are called *patterns*. A finite word $w \in \Sigma^*$ follows a pattern p if there is a way to create w by substituting finite non-empty words in Σ^* for each variable of p.

Example

The word *abab* follows the patterns xx, xy, and $xyzy$, but not xxx.

We are interested in distinguishing patterns that infinite words **must** encounter (some factor follows the pattern) from ones that can be **avoided** (no factor follows the pattern) for some large enough alphabet size.

Patterns We'll Investigate

- Squares (pattern xx), which are 2-unavoidable and 3-avoidable.

Patterns We'll Investigate

- Squares (pattern xx), which are 2-unavoidable and 3-avoidable.
- Cubes (pattern xxx), which are 2-avoidable.

Patterns We'll Investigate

- Squares (pattern xx), which are 2-unavoidable and 3-avoidable.
- Cubes (pattern xxx), which are 2-avoidable.
- Overlaps (pattern $xyxyx$), which are 2-avoidable.

Patterns We'll Investigate

- Squares (pattern xx), which are 2-unavoidable and 3-avoidable.
- Cubes (pattern xxx), which are 2-avoidable.
- Overlaps (pattern $xyxyx$), which are 2-avoidable.
- The Zimin Patterns, which are always unavoidable.

The Thue-Morse Sequence

Let $\mu : \{a, b\}^* \longrightarrow \{a, b\}^*$ be the following morphism:

$$\mu(a) = ab$$
$$\mu(b) = ba$$

The Thue-Morse Sequence

Let $\mu : \{a, b\}^* \longrightarrow \{a, b\}^*$ be the following morphism:

$$\mu(a) = ab$$
$$\mu(b) = ba$$

Consider the words t_n generated by evaluating $\mu^n(a)$:

$$t_0 = a$$
$$t_1 = ab$$
$$t_2 = abba$$
$$t_3 = abbabaab$$
$$t_4 = abbabaabbaababba$$

The Thue-Morse Sequence

Let $\mu : \{a, b\}^* \longrightarrow \{a, b\}^*$ be the following morphism:

$$\mu(a) = ab$$
$$\mu(b) = ba$$

Consider the words t_n generated by evaluating $\mu^n(a)$:

$$t_0 = a$$
$$t_1 = ab$$
$$t_2 = abba$$
$$t_3 = abbabaab$$
$$t_4 = abbabaabbaababba$$

There is an infinite limiting word t, the Thue-Morse Word. Importantly, $\mu(t) = t$.

What Does t Avoid?

What Does t Avoid?

The Thue-Morse Word is cube-free.

What Does t Avoid?

The Thue-Morse Word is cube-free.

Proof (Rough Idea).

1. Assume for contradiction there is a cube uuu of minimal length in t.

What Does t Avoid?

The Thue-Morse Word is cube-free.

Proof (Rough Idea).

1. Assume for contradiction there is a cube uuu of minimal length in t.
2. Use properties of μ to show $|u|$ is even

What Does t Avoid?

The Thue-Morse Word is cube-free.

Proof (Rough Idea).

1. Assume for contradiction there is a cube uuu of minimal length in t.
2. Use properties of μ to show $|u|$ is even
3. Show the factor that would generate uuu after applying μ is a smaller cube in t, contradicting uuu's minimality.

∎

What Does t Avoid?

The Thue-Morse Word is cube-free.

Proof (Rough Idea).

1. Assume for contradiction there is a cube uuu of minimal length in t.
2. Use properties of μ to show $|u|$ is even
3. Show the factor that would generate uuu after applying μ is a smaller cube in t, contradicting uuu's minimality.

∎

The Thue-Morse word is also overlap-free, and a similar proof (albeit with more casework) can prove that this is true.

Using t to Generate a Square-free Ternary Word

We go through the Thue-Morse word and observe how many b's appear between consecutive instances of a. If it is a 0, 1, or 2, we append a, b, or c. Continue infinitely, generating u.

Using t to Generate a Square-free Ternary Word

We go through the Thue-Morse word and observe how many b's appear between consecutive instances of a. If it is a 0, 1, or 2, we append a, b, or c. Continue infinitely, generating u.

Proof that the infinite word u is square-free.

Using t to Generate a Square-free Ternary Word

We go through the Thue-Morse word and observe how many b's appear between consecutive instances of a. If it is a 0, 1, or 2, we append a, b, or c. Continue infinitely, generating u.

Proof that the infinite word u is square-free.

1. Define the morphism $\pi : a \to a, b \to ab, c \to abb$, and notice $\pi(u) = t$.

Using t to Generate a Square-free Ternary Word

We go through the Thue-Morse word and observe how many b's appear between consecutive instances of a. If it is a 0, 1, or 2, we append a, b, or c. Continue infinitely, generating u.

Proof that the infinite word u is square-free.

1. Define the morphism $\pi : a \to a, b \to ab, c \to abb$, and notice $\pi(u) = t$.
2. Assume a square vv exists in u for the sake of contradiction. Consider vv and whatever letter appears next, say x.

Using t to Generate a Square-free Ternary Word

We go through the Thue-Morse word and observe how many b's appear between consecutive instances of a. If it is a 0, 1, or 2, we append a, b, or c. Continue infinitely, generating u.

Proof that the infinite word u is square-free.

1. Define the morphism $\pi : a \to a, b \to ab, c \to abb$, and notice $\pi(u) = t$.
2. Assume a square vv exists in u for the sake of contradiction. Consider vv and whatever letter appears next, say x.
3. It follows $\pi(vvx) = \pi(v)\pi(v)\pi(x)$ appears in t, and since each of these factors starts with an a, $\pi(vvx)$ can be rewritten as $aw_1aw_1aw_2$, which implies the existence of either a cube or overlap in t, a contradiction.

∎

Unavoidable Patterns

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Definition

Start with $Z_0 = \varepsilon$, and perform the following process:

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Definition

Start with $Z_0 = \varepsilon$, and perform the following process:
Given the pattern Z_n, we pick a variable x in Δ that has not been used prior and perform the concatenation $Z_{n+1} = Z_n x Z_n$. Here are the first few examples:

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Definition

Start with $Z_0 = \varepsilon$, and perform the following process:
Given the pattern Z_n, we pick a variable x in Δ that has not been used prior and perform the concatenation $Z_{n+1} = Z_n x Z_n$. Here are the first few examples:

- $Z_1 = \alpha$

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Definition

Start with $Z_0 = \varepsilon$, and perform the following process:
Given the pattern Z_n, we pick a variable x in Δ that has not been used prior and perform the concatenation $Z_{n+1} = Z_n x Z_n$. Here are the first few examples:

- $Z_1 = \alpha$
- $Z_2 = \alpha\beta\alpha$

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Definition

Start with $Z_0 = \varepsilon$, and perform the following process:
Given the pattern Z_n, we pick a variable x in Δ that has not been used prior and perform the concatenation $Z_{n+1} = Z_n x Z_n$. Here are the first few examples:

- $Z_1 = \alpha$
- $Z_2 = \alpha\beta\alpha$
- $Z_3 = \alpha\beta\alpha\gamma\alpha\beta\alpha$

Defining the Zimin Patterns

We once again consider the alphabet of variables Δ. The *Zimin patterns* are defined as follows:

Definition

Start with $Z_0 = \varepsilon$, and perform the following process:
Given the pattern Z_n, we pick a variable x in Δ that has not been used prior and perform the concatenation $Z_{n+1} = Z_n x Z_n$. Here are the first few examples:

- $Z_1 = \alpha$
- $Z_2 = \alpha\beta\alpha$
- $Z_3 = \alpha\beta\alpha\gamma\alpha\beta\alpha$

It turns out all Zimin patterns are unavoidable, no matter how large the alphabet is.

Proving Unavoidability

Proposition

All Zimin patterns are unavoidable on all alphabets.

Proving Unavoidability

Proposition

All Zimin patterns are unavoidable on all alphabets.

Proof.

We proceed by induction, assuming that Z_n is encountered by all words of some finite length l in Σ^*.

Proving Unavoidability

Proposition

All Zimin patterns are unavoidable on all alphabets.

Proof.

We proceed by induction, assuming that Z_n is encountered by all words of some finite length l in Σ^*.

1. Base case: $Z_1 = \alpha$ is unavoidable for all words of length 1.

Proving Unavoidability

Proposition

All Zimin patterns are unavoidable on all alphabets.

Proof.

We proceed by induction, assuming that Z_n is encountered by all words of some finite length l in Σ^*.

1. Base case: $Z_1 = \alpha$ is unavoidable for all words of length 1.
2. Inductive proof: Split into blocks of l letters, with one letter of space between each. After a large enough number of blocks, we must have a block repeated.

Proving Unavoidability

Proposition

All Zimin patterns are unavoidable on all alphabets.

Proof.

We proceed by induction, assuming that Z_n is encountered by all words of some finite length l in Σ^*.

1. Base case: $Z_1 = \alpha$ is unavoidable for all words of length 1.
2. Inductive proof: Split into blocks of l letters, with one letter of space between each. After a large enough number of blocks, we must have a block repeated. The blocks are identical and must contain the same word following Z_n. We set the new variable to be the word between the instances of Z_n. Thus, we have encountered Z_{n+1} within a finite number of letters.

■

Zimin's Characterization

Theorem (Zimin's Theorem)

A pattern p is unavoidable on all alphabets if and only if p is a factor of a Zimin pattern.

The proof of this is omitted, but results from reducing patterns via the Zimin algorithm.

Some Open Problems

Extremal Words

Definition

We say a finite word is *extremal pattern-free* if the word avoids the pattern, but all *extensions* (additions of a letter to the word at any position) cause it to encounter the pattern.

Extremal Words

Definition

We say a finite word is *extremal pattern-free* if the word avoids the pattern, but all *extensions* (additions of a letter to the word at any position) cause it to encounter the pattern.

Example

The only binary extremal square-free words are *aba* and *bab*.

Extremal Words

Definition

We say a finite word is *extremal pattern-free* if the word avoids the pattern, but all *extensions* (additions of a letter to the word at any position) cause it to encounter the pattern.

Example

The only binary extremal square-free words are *aba* and *bab*.

Example

There are infinitely many ternary extremal square-free words, the shortest of which is *abcabacbcabcbabcabacbcabc*.

Open Questions

- Two MIT students proved in 2021 that there are no extremal square-free words for alphabets of size 17 or larger.

Open Questions

- Two MIT students proved in 2021 that there are no extremal square-free words for alphabets of size 17 or larger.
- But are there extremal square-free words for alphabets of size 4-16?

Open Questions

- Two MIT students proved in 2021 that there are no extremal square-free words for alphabets of size 17 or larger.
- But are there extremal square-free words for alphabets of size 4-16?
- Are there any extremal cube-free binary words?

Open Questions

- Two MIT students proved in 2021 that there are no extremal square-free words for alphabets of size 17 or larger.
- But are there extremal square-free words for alphabets of size 4-16?
- Are there any extremal cube-free binary words?
- It is known that infinitely many extremal overlap-free words exist in binary alphabets, but do any exist for larger alphabets?

Thank you!