The Mian Pricinple Benhid Erorr-Corrceiton Techquines

Christian Häger

Department of Signals and Systems
Communication Systems Group
Chalmers University of Technology
Gothenburg, Sweden

May 21, 2012

CHALMERS

Learning Objectives

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm.

Learning Objectives

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm.

After this lecture, you should be able to:

Learning Objectives

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm.

After this lecture, you should be able to:

- Apply the error-correction principle and explain why we can read the text.

Learning Objectives

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm.

After this lecture, you should be able to:

- Apply the error-correction principle and explain why we can read the text.
- Critically evaluate the hypothesis in the second sentence based on what you will learn about the fundamental limitation of error-correction.

Introduction

Introduction

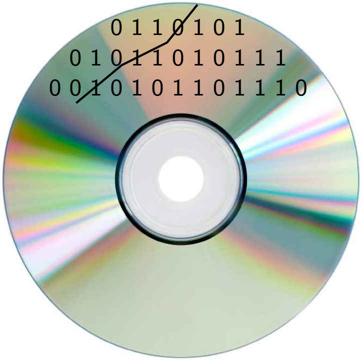
written on the CD: . . . 0 1 0 1 1 0 1 0 1 . . .

Introduction

0 1 1 0 1 0 1
0 1 0 1 1 0 1 0 1 1 1
0 0 1 0 1 0 1 1 0 1 1 1 0

written on the CD: . . . 0 1 0 1 1 0 1 0 1 . . .

Introduction



written on the CD: . . . 0 1 0 1 1 0 1 0 1 . . .
after the scratch: . . . 0 1 1 1 1 0 0 0 1 . . .

Introduction

0 1 1 0 1 0 1
0 1 0 1 1 0 1 0 1 1 1
0 0 1 0 1 0 1 1 0 1 1 1 0

written on the CD: $\ldots 0\ 1\ \boxed{0}\ 1\ 1\ 0\ \boxed{1}\ 0\ 1 \ldots$

after the scratch: $\ldots 0\ 1\ \boxed{1}\ 1\ 1\ 0\ \boxed{0}\ 0\ 1 \ldots$

bit errors

Introduction

0 1 1 0 1 0 1
0 1 0 1 1 0 1 0 1 1 1
0 0 1 0 1 0 1 1 0 1 1 1 0

written on the CD: $\dots 0\ 1\ \boxed{0}\ 1\ 1\ 0\ \boxed{1}\ 0\ 1 \dots$

after the scratch: $\dots 0\ 1\ \boxed{1}\ 1\ 1\ 0\ \boxed{0}\ 0\ 1 \dots$

bit errors

What do we do now?

The engineering goal is to provide (almost) error-free performance, even in the presence of scratches.

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1
- You can only write bits on the CD

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!

$$1\ 0\ 1 \xrightarrow{\quad ? \quad}$$

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

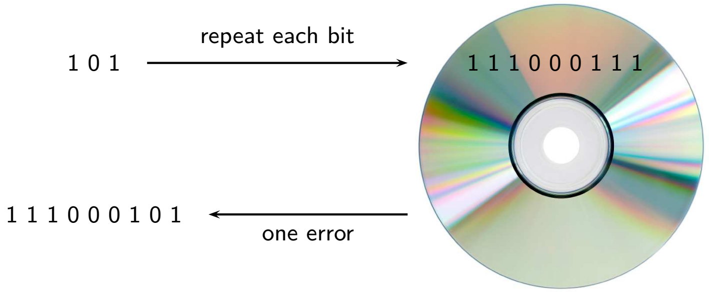
- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!

$$1\ 0\ 1 \xrightarrow{\text{repeat each bit}} 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1$$

Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

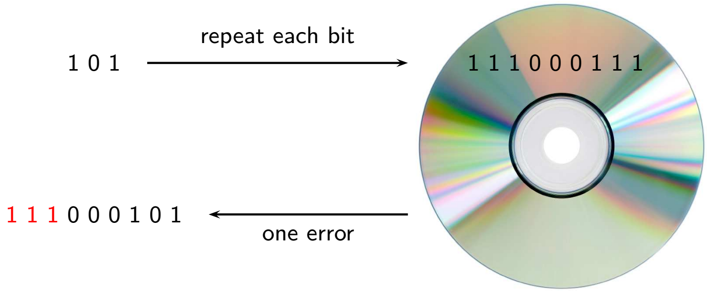
- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!



Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

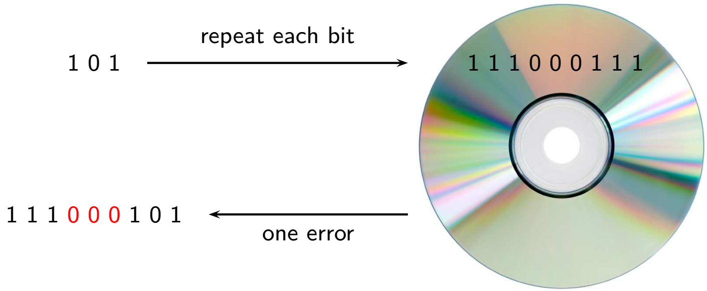
- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!



Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

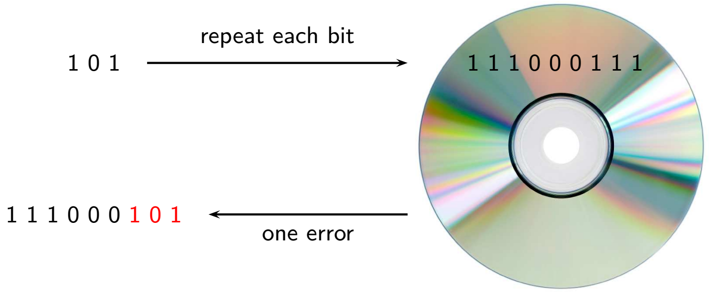
- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!



Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!



Short Assignment (1 Minute)

Can you think of a simple way to solve the problem?

- Assume that the data is 1 0 1
- You can only write bits on the CD
- One bit will be flipped after a scratch.
- Don't try to be efficient! You may write as much on the CD as you want.
- What would you write on the CD? Discuss with your neighbor!

$$1\ 0\ 1 \xrightarrow{\text{repeat each bit}} 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1$$

looks more like 1 1 1 than 0 0 0

$$1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1 \xleftarrow[\text{one error}]{}$$

The Main Principle

The Main Principle

1.

2.

The Main Principle

1. We constructed a code

2.

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.

2.

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011,\ 100, 101, 110, 111\}$

2.

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011, 100, 101, 110, 111\}$
- after $= \{000, 111\}$

2.

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011, 100, 101, 110, 111\}$
- after $= \{000, 111\}$
- This is called a code: A subset of an initially larger set of bit-words

2.

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011, 100, 101, 110, 111\}$
- after $= \{000, 111\}$
- This is called a code: A subset of an initially larger set of bit-words

2. We intuitively used a distance function d(x, y)

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011, 100, 101, 110, 111\}$
- after $= \{000, 111\}$
- This is called a code: A subset of an initially larger set of bit-words

2. We intuitively used a distance function d(x, y)

- Distance between bit-words: Count the number of different bits

The Main Principle

1. We constructed a code

- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011, \ 100, 101, 110, 111\}$
- after $= \{000, 111\}$
- This is called a code: A subset of an initially larger set of bit-words

2. We intuitively used a distance function d(x, y)

- Distance between bit-words: Count the number of different bits
- d$(001, 011) = 1$

The Main Principle

1. We constructed a code

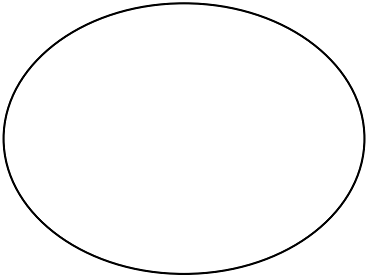
- Bit-words: 3 consecutive bits.
- before $= \{000, 001, 010, 011, 100, 101, 110, 111\}$
- after $= \{000, 111\}$
- This is called a code: A subset of an initially larger set of bit-words

2. We intuitively used a distance function $\mathsf{d}(x, y)$

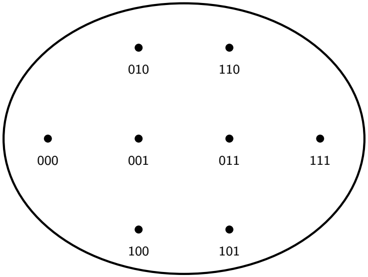
- Distance between bit-words: Count the number of different bits
- $\mathsf{d}(001, 011) = 1$
- $\mathsf{d}(000, 111) = 3$

Graphical Interpretation

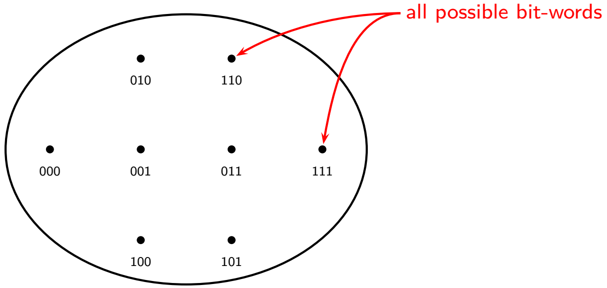
Graphical Interpretation



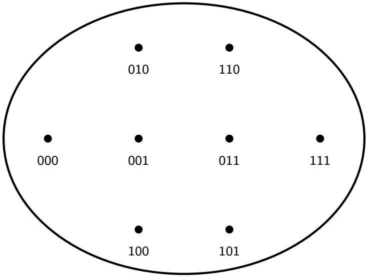
Graphical Interpretation



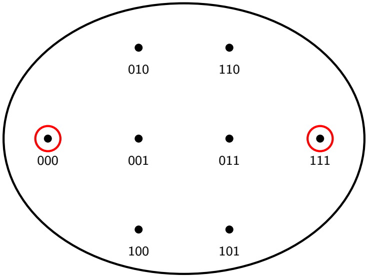
Graphical Interpretation



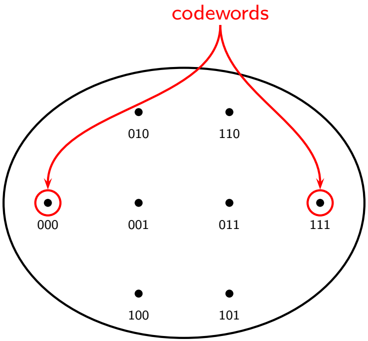
Graphical Interpretation



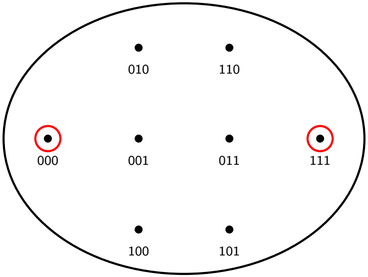
Graphical Interpretation



Graphical Interpretation

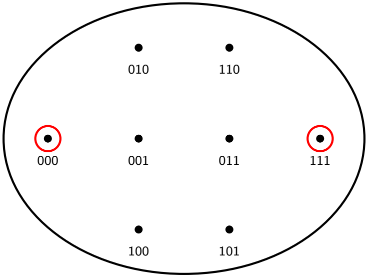


Graphical Interpretation



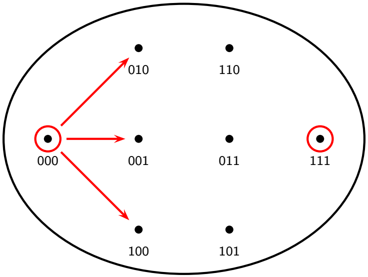
Graphical Interpretation

What happens in the case of an error?

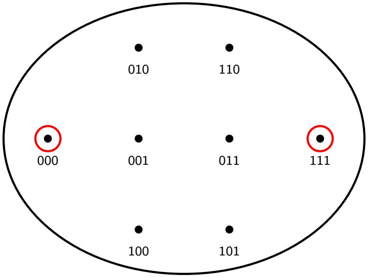


Graphical Interpretation

What happens in the case of an error?

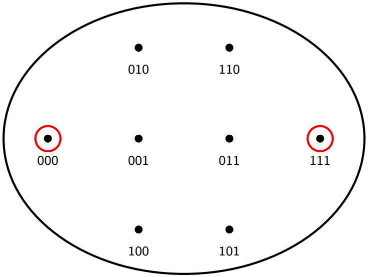


Graphical Interpretation



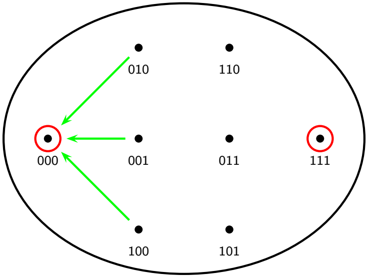
Graphical Interpretation

What is decoding?

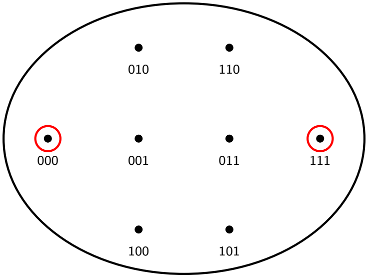


Graphical Interpretation

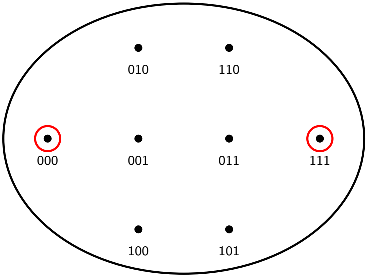
What is decoding?



Graphical Interpretation



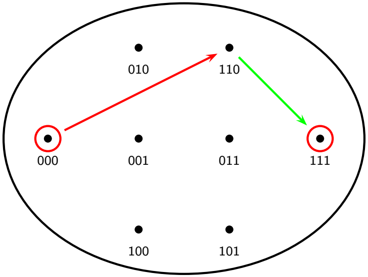
Graphical Interpretation



Fundamental Limitation of Error-Correction

An error cannot be corrected if it is so large that one moves too far away from the true codeword and is now, in fact, closer to another codeword.

Graphical Interpretation



Fundamental Limitation of Error-Correction

An error cannot be corrected if it is so large that one moves too far away from the true codeword and is now, in fact, closer to another codeword.

How does this apply to language?

How does this apply to language?

- Language can be seen as an error-correcting code.

How does this apply to language?

- Language can be seen as an error-correcting code.
- Not all 5-letter combinations are words: “akzke” and “plfee” are not words, but “house” is.

How does this apply to language?

- Language can be seen as an error-correcting code.
- Not all 5-letter combinations are words: “akzke” and “plfee” are not words, but “house” is.

Assumptions

How does this apply to language?

- Language can be seen as an error-correcting code.
- Not all 5-letter combinations are words: “akzke” and “plfee” are not words, but “house” is.

Assumptions

- Simplicity: We will focus on words (not sentences).

How does this apply to language?

- Language can be seen as an error-correcting code.
- Not all 5-letter combinations are words: “akzke” and “plfee” are not words, but “house” is.

Assumptions

- Simplicity: We will focus on words (not sentences).
- Shuffling letters = errors

How does this apply to language?

- Language can be seen as an error-correcting code.
- Not all 5-letter combinations are words: “akzke” and “plfee” are not words, but “house” is.

Assumptions

- Simplicity: We will focus on words (not sentences).
- Shuffling letters = errors
- The brain acts as a decoder.

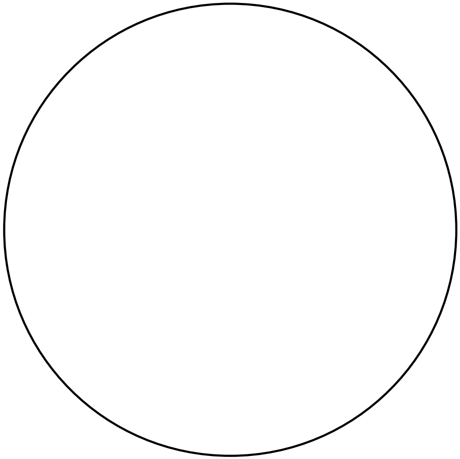
How does this apply to language?

- Language can be seen as an error-correcting code.
- Not all 5-letter combinations are words: “akzke” and “plfee” are not words, but “house” is.

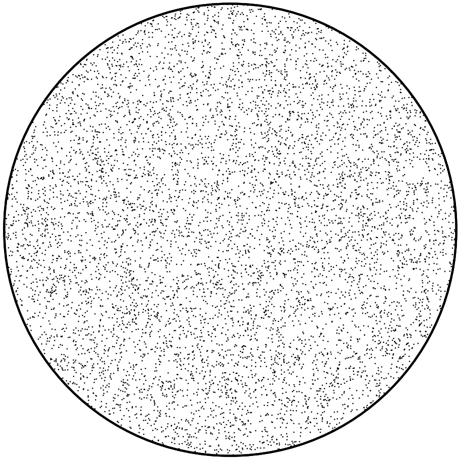
Assumptions

- Simplicity: We will focus on words (not sentences).
- Shuffling letters = errors
- The brain acts as a decoder.
- A distance function exists (but may be hard to define rigorously).

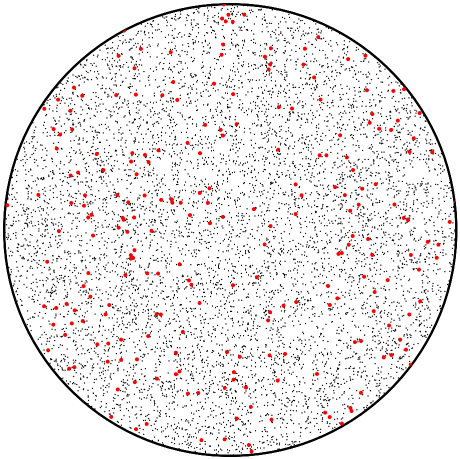
The Space of All 5-Letter Combinations



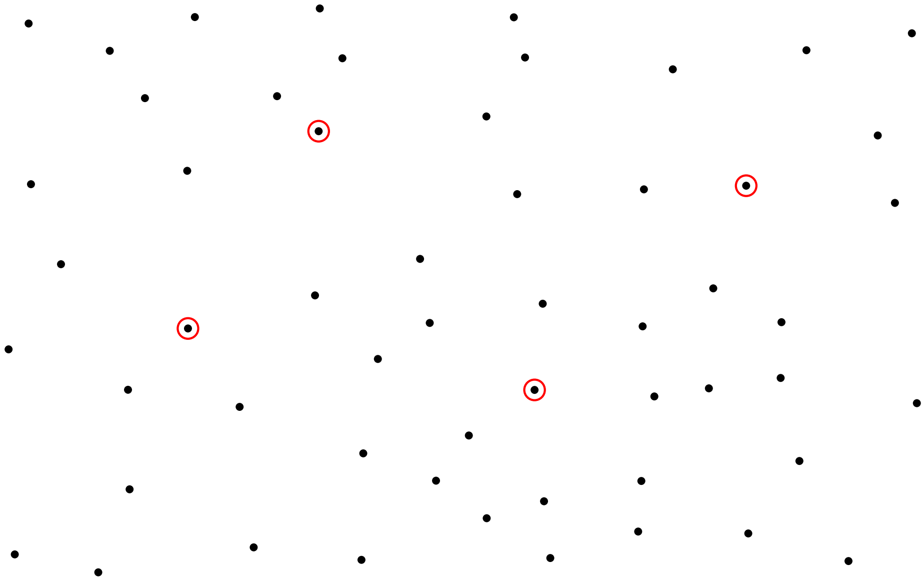
The Space of All 5-Letter Combinations



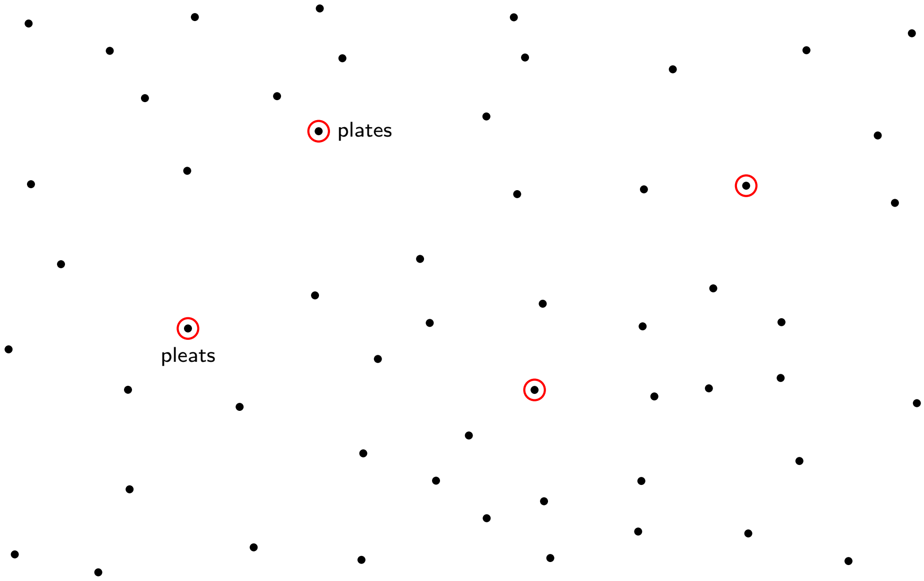
The Space of All 5-Letter Combinations



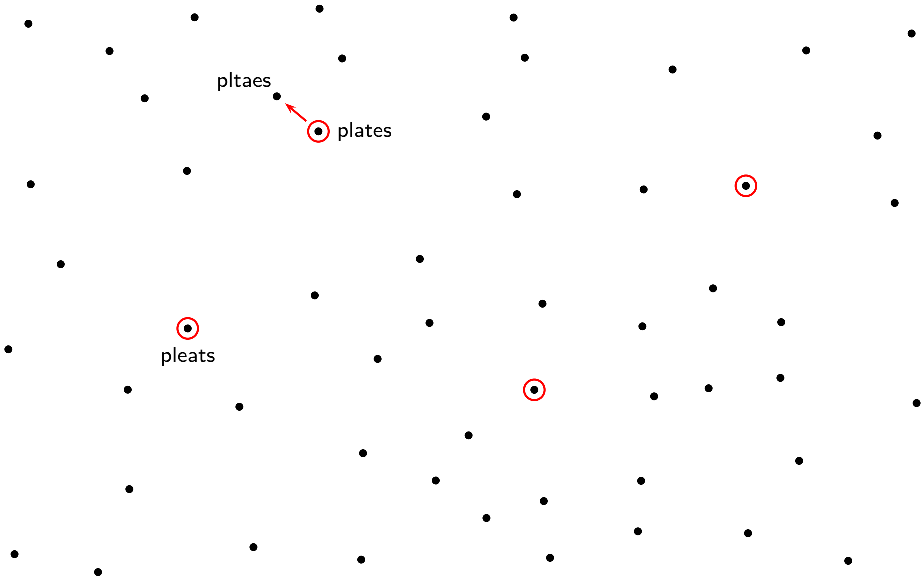
Close-Up View



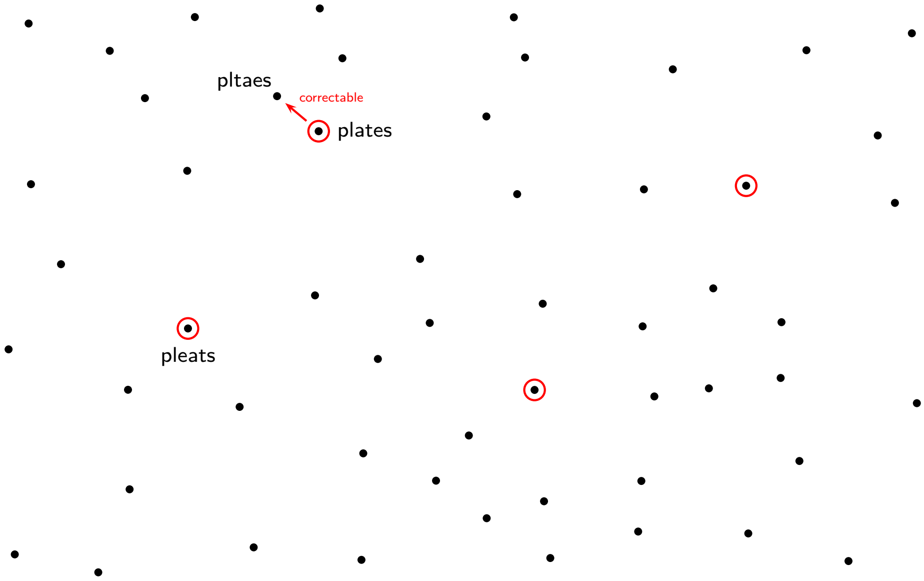
Close-Up View



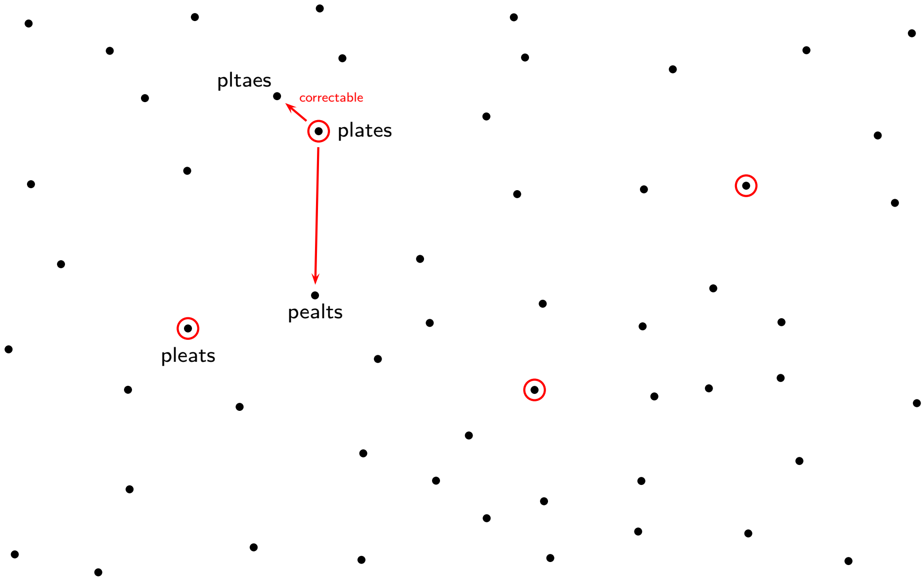
Close-Up View



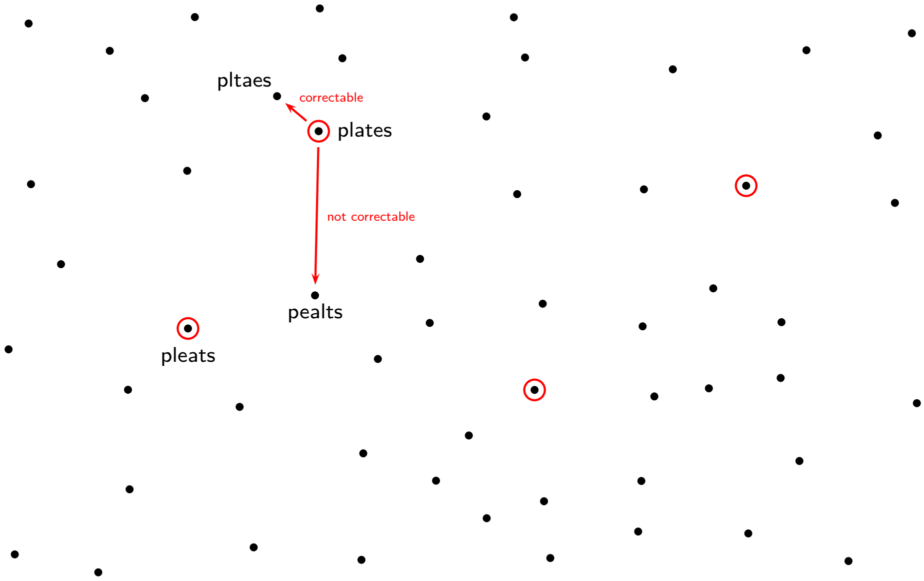
Close-Up View



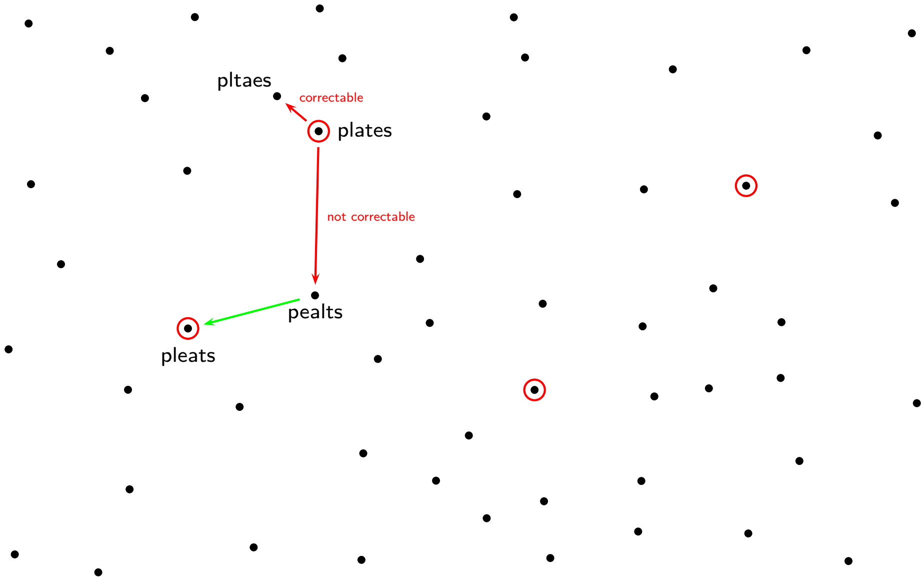
Close-Up View



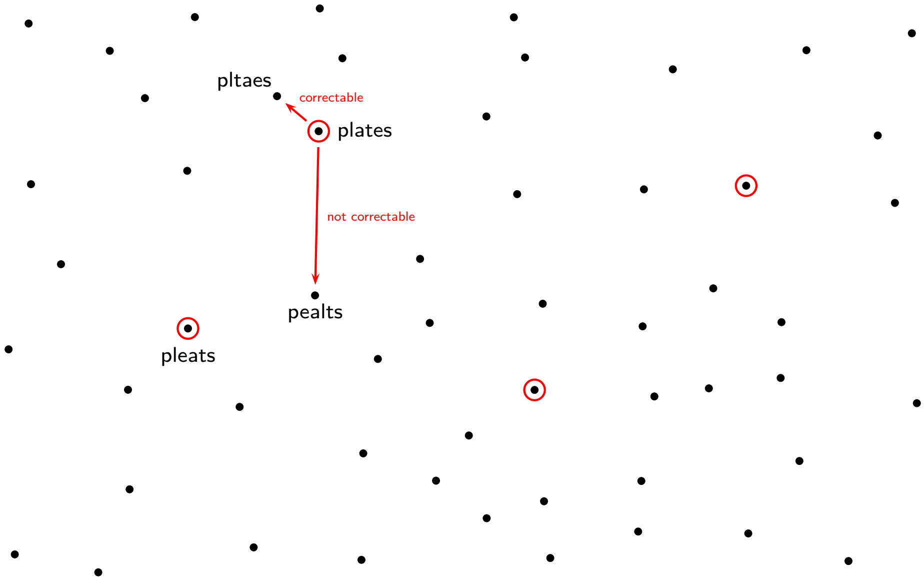
Close-Up View



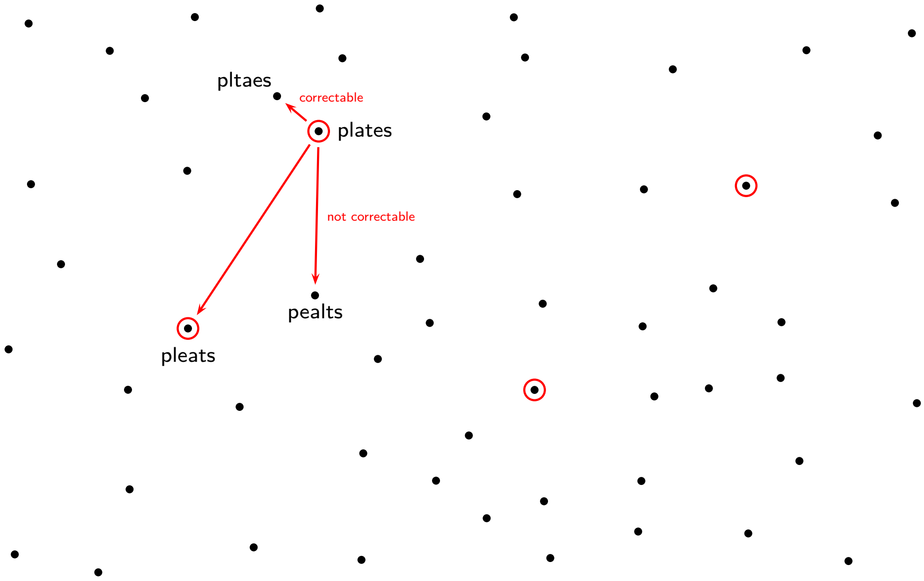
Close-Up View



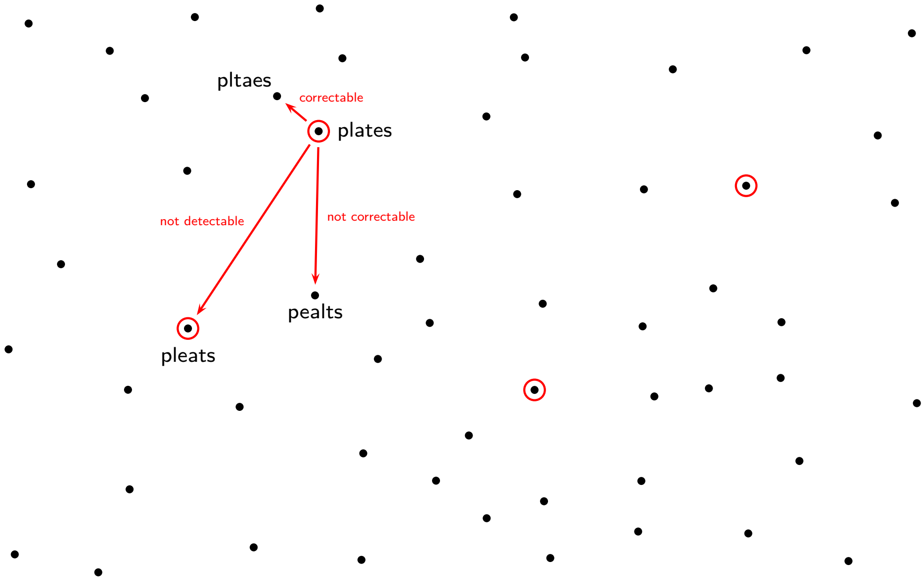
Close-Up View



Close-Up View



Close-Up View



Summary

Summary

- Error-correction requires two things: .

Summary

- Error-correction requires two things: a code .

Summary

- Error-correction requires two things: a code and a distance function.

Summary

- Error-correction requires two things: a code and a distance function.
- Error-correction has limits based on the properties of the code.

Summary

- Error-correction requires two things: a code and a distance function.
- Error-correction has limits based on the properties of the code.
- Language can be interpreted as a code.

Learning Objectives

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm.

Are you able to:

Learning Objectives

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm.

Are you able to:

- Apply the error-correction principle and explain why we can read the text?
- Critically evaluate the hypothesis in the second sentence based on what you learned about the fundamental limitation of error-correction?