Lecture 2: Writers’ aids: Spelling errors

LING-351 Language Technology and LLMs

Instructor: Hakyung Sung

August 28, 2025

*Acknowledgment: These course slides are based on materials by Lelia Glass @ Georgia Tech (Course: Language & Computers)

Table of contents

Review

Language, writing, encoding

- Language

Language, writing, encoding

- Language
- Writing

Language, writing, encoding

- Language
- Writing
- Language = writing?

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:
    - Alphabetic

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:
    - Alphabetic $\rightarrow$ sounds

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:
    - Alphabetic $\rightarrow$ sounds
    - Syllabic

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:
    - Alphabetic $\rightarrow$ sounds
    - Syllabic $\rightarrow$ syllables

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:
    - Alphabetic $\rightarrow$ sounds
    - Syllabic $\rightarrow$ syllables
    - Logographic

Language, writing, encoding

- Language
- Writing
- Language = writing?
- Three major systems to encode languages:
    - Alphabetic $\rightarrow$ sounds
    - Syllabic $\rightarrow$ syllables
    - Logographic $\rightarrow$ meanings

Then, what does digital writing encode?

- Digital technology can be understood as another form of writing to encode language into digital formats

Then, what does digital writing encode?

- Digital technology can be understood as another form of writing to encode language into digital formats
- Bit (0/1 signal): the smallest unit of digital information

Then, what does digital writing encode?

- Digital technology can be understood as another form of writing to encode language into digital formats
- Bit (0/1 signal): the smallest unit of digital information
- Byte (8 bits): a bundle of 8 bits, the basic unit of storage

Then, what does digital writing encode?

- Digital technology can be understood as another form of writing to encode language into digital formats
- Bit (0/1 signal): the smallest unit of digital information
- Byte (8 bits): a bundle of 8 bits, the basic unit of storage
- Character encoding (UTF-8): rules that map bytes to code points

Lesson plan

Lesson plan

- ~~Review~~
- Spelling problems in writing

Lesson plan

- ~~Review~~
- Spelling problems in writing
- Different types of spelling errors

Lesson plan

- ~~Review~~
- Spelling problems in writing
- Different types of spelling errors
- Building a simple spell-checker

Lesson plan

- ~~Review~~
- Spelling problems in writing
- Different types of spelling errors
- Building a simple spell-checker
- Thinking about a more complex spell-checker

Lesson plan

- ~~Review~~
- Spelling problems in writing
- Different types of spelling errors
- Building a simple spell-checker
- Thinking about a more complex spell-checker
- Wrap-up

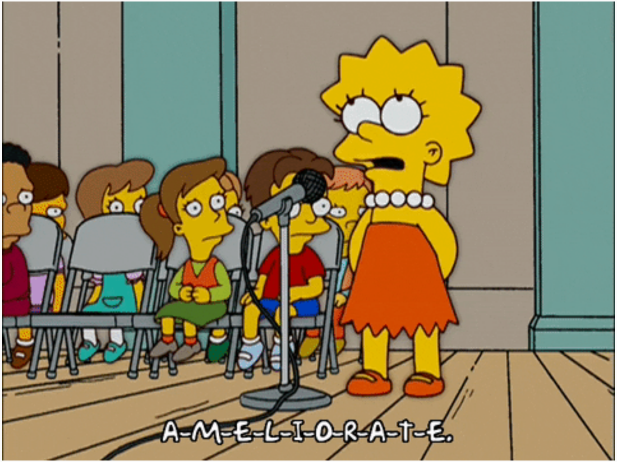
Key idea: ~~Spelling errors are annoying~~

Lesson plan

- ~~Review~~
- Spelling problems in writing
- Different types of spelling errors
- Building a simple spell-checker
- Thinking about a more complex spell-checker
- Wrap-up

Key idea: ~~Spelling errors are annoying~~
Spelling errors vary by types (and even by languages);
there is no one-size-fits-all solution.

Spelling problems in writing


A-M-E-L-I-O-R-A-T-E.

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
    - Before that, Old English was written in the runic script

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
    - Before that, Old English was written in the runic script
    - Christian missionaries introduced the Latin alphabet in the 7th century

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system
- Related technologies:

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system
- Related technologies:
  - Handwriting on parchment and paper (monastic scribes, medieval manuscripts)

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system
- Related technologies:
  - Handwriting on parchment and paper (monastic scribes, medieval manuscripts)
  - Printing press (1470s in Englan)
    $\rightarrow$ wider literacy, circulation of books

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system
- Related technologies:
  - Handwriting on parchment and paper (monastic scribes, medieval manuscripts)
  - Printing press (1470s in Englan)
    $\rightarrow$ wider literacy, circulation of books
  - Typewriter (1860s)
    $\rightarrow$ faster, more uniform writing

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system
- Related technologies:
  - Handwriting on parchment and paper (monastic scribes, medieval manuscripts)
  - Printing press (1470s in Englan)
    $\rightarrow$ wider literacy, circulation of books
  - Typewriter (1860s)
    $\rightarrow$ faster, more uniform writing
  - Digital word processing (20th century)
    $\rightarrow$ autocorrect, spell checkers

Writing English: A long evolution

- English has used the Latin alphabet since the 9th century
  - Before that, Old English was written in the runic script
  - Christian missionaries introduced the Latin alphabet in the 7th century
  - By the 9th century, it became the dominant writing system
- Related technologies:
  - Handwriting on parchment and paper (monastic scribes, medieval manuscripts)
  - Printing press (1470s in Englan)
    $\rightarrow$ wider literacy, circulation of books
  - Typewriter (1860s)
    $\rightarrow$ faster, more uniform writing
  - Digital word processing (20th century)
    $\rightarrow$ autocorrect, spell checkers
- Standardized spelling came much later...

English without standard spelling

- Spelling wasn’t standardized until the mid-1600s to 1700s

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:
  - **King James Bible (1611)** – named after King James I of England, who authorized a new translation by scholars to unify religious practices

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:
  - **King James Bible (1611)** – named after King James I of England, who authorized a new translation by scholars to unify religious practices
  - **Early dictionaries** – e.g., Robert Cawdrey's *Table Alphabeticall* (1604), Samuel Johnson's dictionary (1755)

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:
  - **King James Bible (1611)** – named after King James I of England, who authorized a new translation by scholars to unify religious practices
  - **Early dictionaries** – e.g., Robert Cawdrey's *Table Alphabeticall* (1604), Samuel Johnson's dictionary (1755)
- Authors themselves didn't use consistent spelling

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:
  - **King James Bible (1611)** – named after King James I of England, who authorized a new translation by scholars to unify religious practices
  - **Early dictionaries** – e.g., Robert Cawdrey's *Table Alphabeticall* (1604), Samuel Johnson's dictionary (1755)
- Authors themselves didn't use consistent spelling
- Shakespeare's name appeared in many forms:

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:
    - **King James Bible (1611)** – named after King James I of England, who authorized a new translation by scholars to unify religious practices
    - **Early dictionaries** – e.g., Robert Cawdrey's *Table Alphabeticall* (1604), Samuel Johnson's dictionary (1755)
- Authors themselves didn't use consistent spelling
- Shakespeare's name appeared in many forms:

English without standard spelling

- Spelling wasn't standardized until the mid-1600s to 1700s
- Influenced by:
    - **King James Bible (1611)** – named after King James I of England, who authorized a new translation by scholars to unify religious practices
    - **Early dictionaries** – e.g., Robert Cawdrey's *Table Alphabeticall* (1604), Samuel Johnson's dictionary (1755)
- Authors themselves didn't use consistent spelling
- Shakespeare's name appeared in many forms:

Willm Shakp, William Shaksper, Wm Shakspe, William Shakspere, Willm Shakspere, William Shakspeare

Why *standardized* spelling?

- Even without standard spelling, we understand:

Why *standardized* spelling?

- Even without standard spelling, we understand:

  To what extent do the ***spellling errers*** *in this* ***setnence dirsupt*** *your* ***undertsanding****?*

Why *standardized* spelling?

- Even without standard spelling, we understand:

  To what extent do the ***spellling errers*** *in this* ***setnence dirsupt*** *your* ***undertsanding****?*

- Readers often focus on **word shape**, not letter-by-letter decoding

What if everyone spelled freely?

- In Shakespeare’s time, spelling was flexible.

What if everyone spelled freely?

- In Shakespeare's time, spelling was flexible.
- Imagine replacing English spelling with IPA (phonetic spelling).

What if everyone spelled freely?

- In Shakespeare's time, spelling was flexible.
- Imagine replacing English spelling with **IPA** (phonetic spelling).

Question

What are the benefits and drawbacks of having a standardized spelling system?

Benefits of standardized spelling

- Supports literacy across dialects/various pronunciations (e.g., *tomato*, *Atlanta*)

Benefits of standardized spelling

- Supports literacy across dialects/various pronunciations (e.g., *tomato*, *Atlanta*)
- Enables searching and record-keeping

Spelling in the digital age

- English spelling remains complex

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
    - Spell checkers

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
    - Spell checkers
    - Predictive text; auto-complete

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
    - Spell checkers
    - Predictive text; auto-complete
    - Generative LLMs

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
  - Spell checkers
  - Predictive text; auto-complete
  - Generative LLMs

Group discussion

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
    - Spell checkers
    - Predictive text; auto-complete
    - Generative LLMs

Group discussion

- *(Put into the shared deck)* Come up with at least one example

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
  - Spell checkers
  - Predictive text; auto-complete
  - Generative LLMs

Group discussion

- *(Put into the shared deck)* Come up with at least one example
- How often do you use tools to check the spelling errors?

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
  - Spell checkers
  - Predictive text; auto-complete
  - Generative LLMs

Group discussion

- *(Put into the shared deck)* **Come up with at least one example**
- How often do you use tools to check the spelling errors?
- Which one do you rely on the most?

Spelling in the digital age

- English spelling remains complex
- Thankfully, we have **writers' aids**:
  - Spell checkers
  - Predictive text; auto-complete
  - Generative LLMs

Group discussion

- *(Put into the shared deck)* **Come up with at least one example**
- How often do you use tools to check the spelling errors?
- Which one do you rely on the most?
- Do they ever create problems (instead of solving them)?

Breaking down the problem

Not all spelling errors are the same.

Breaking down the problem

Not all spelling errors are the same.
To solve them, we need to consider **error types**.

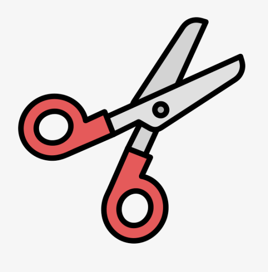
Different types of spelling errors

Spelling error types

- 1. Non-word errors
- 2. Real-word errors
- *Notes.* How common?

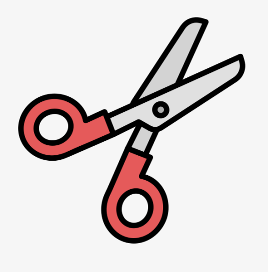
1. Non-word errors

- True confusion:



1. Non-word errors

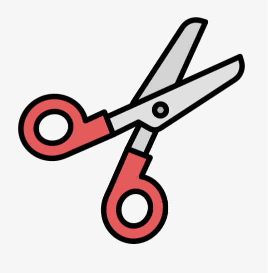
- True confusion:



- “*sissors*” (not knowing the correct form)

1. Non-word errors

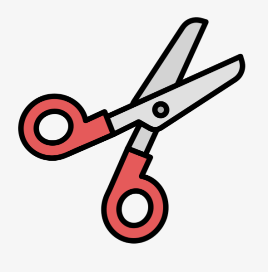
- True confusion:



- "*sissors*" (not knowing the correct form)
- Typos: "*hte*" (keyboard slip)

1. Non-word errors

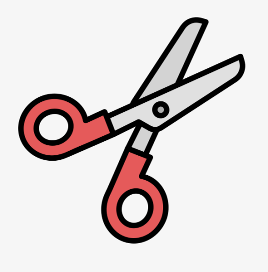
- **True confusion**:



- "*sissors*" (not knowing the correct form)
- **Typos**: "*hte*" (keyboard slip)
- Automatically detected when:

1. Non-word errors

- **True confusion**:



- “*sissors*” (not knowing the correct form)
- **Typos**: “*hte*” (keyboard slip)
- Automatically detected when:
  - Word *not found* in **dictionary** of correct spellings

Quantifying misspellings: Edit distance

- Measures how “far apart” two strings are

Quantifying misspellings: Edit distance

- Measures how “far apart” two strings are
- Known as **Levenshtein distance**

Quantifying misspellings: Edit distance

- Measures how “far apart” two strings are
- Known as **Levenshtein distance**
- Minimum number of operations to transform one word into another

Edit distance: Basic operations

Each operation = 1 unit of cost

- **Insertion**: *aquire* → *ac<u>c</u>quire*
- **Deletion**: *argu<u>e</u>ment* → *argument*
- **Substitution**: *cal<u>e</u>nder* → *cal<u>a</u>ndar*
- **Transposition**: *con<u>cs</u>ious* → *con<u>sc</u>ious*
    - Sometimes counted as two substitutions

Why does edit distance matter?

- Helps suggest the **closest correct word** when a typo is found

Why does edit distance matter?

- Helps suggest the **closest correct word** when a typo is found
- In other words, can be used to suggest candidate corrections

Why does edit distance matter?

- Helps suggest the **closest correct word** when a typo is found
- In other words, can be used to suggest candidate corrections
    1. Input: *recieve*

Why does edit distance matter?

- Helps suggest the **closest correct word** when a typo is found
- In other words, can be used to suggest candidate corrections
    1. Input: *recieve*
        - Candidates: *receive, recipe*

Why does edit distance matter?

- Helps suggest the **closest correct word** when a typo is found
- In other words, can be used to suggest candidate corrections
    1. Input: *recieve*
        - Candidates: *receive*, *recipe*
    2. Input: *acommodation*

Why does edit distance matter?

- Helps suggest the **closest correct word** when a typo is found
- In other words, can be used to suggest candidate corrections
  1. Input: *recieve*
     - Candidates: *receive, recipe*
  2. Input: *acommodation*
     - Candidates: *accommodation, commendation*

Weighted edit distance

- Not all errors are equally likely

Weighted edit distance

- Not all errors are equally likely
- Edit distance can be **weighted** for more realistic corrections

Weighted edit distance

- Not all errors are equally likely
- Edit distance can be **weighted** for more realistic corrections
- Substituting a nearby key on the keyboard may cost less than a distant one

Weighted edit distance

- Not all errors are equally likely
- Edit distance can be **weighted** for more realistic corrections
- Substituting a nearby key on the keyboard may cost less than a distant one
  - e.g., *friemd* $\rightarrow$ *friend* (substitution: *m*→*n*, keys are adjacent → low cost)

Weighted edit distance

- Not all errors are equally likely
- Edit distance can be **weighted** for more realistic corrections
- Substituting a nearby key on the keyboard may cost less than a distant one
    - e.g., *friemd* $\rightarrow$ *friend* (substitution: *m*→*n*, keys are adjacent → low cost)
    - vs. *friemd* $\rightarrow$ *fried* (deletion of *m*, more disruptive → higher cost)

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words

Limitations

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates

Limitations

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates
- Suggests the closest candidate as the correction

Limitations

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates
- Suggests the closest candidate as the correction
- Adds some weights for more realistic correction

Limitations

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates
- Suggests the closest candidate as the correction
- Adds some weights for more realistic correction

Limitations

- Fails with new words or domain-specific terms (e.g., *rizz*, *COVID-19*)

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates
- Suggests the closest candidate as the correction
- Adds some weights for more realistic correction

Limitations

- Fails with new words or domain-specific terms (e.g., *rizz*, *COVID-19*)
- Ignores context (e.g., *I want to by a book* → intended: *buy*)

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates
- Suggests the closest candidate as the correction
- Adds some weights for more realistic correction

Limitations

- Fails with new words or domain-specific terms (e.g., *rizz*, *COVID-19*)
- Ignores context (e.g., *I want to by a book* → intended: *buy*)

Interim summary

Traditional method: Dictionary + Edit Distance: How it works

- Relies on a dictionary of correct words
- Calculates distance between misspelling and candidates
- Suggests the closest candidate as the correction
- Adds some weights for more realistic correction

Limitations

- Fails with new words or domain-specific terms (e.g., *rizz*, *COVID-19*)
- Ignores context (e.g., *I want to by a book* → intended: *buy*)

Q. What happens if the misspelled word is still a real word?

1. Local syntactic errors: **Their* was a problem*

2. Real-word Errors

1. Local syntactic errors: **Their* was a problem*
2. Long-distance syntactic errors: **The key to the cabinets are on the table*

2. Real-word Errors

1. Local syntactic errors: **Their was a problem*
2. Long-distance syntactic errors: **The key to the cabinets are on the table*
3. Semantic errors: *I read the brook*

2. Real-word Errors

1. Local syntactic errors: **<u>Their</u> was a problem*
2. Long-distance syntactic errors: **The key to the cabinets <u>are</u> on the table*
3. Semantic errors: *I read the <u>brook</u>*

- Solving this problem is more difficult:

2. Real-word Errors

1. Local syntactic errors: **Their* was a problem*
2. Long-distance syntactic errors: **The key to the cabinets are on the table*
3. Semantic errors: *I read the brook*

- Solving this problem is more difficult:
    - The result is still a valid word → not flagged by a dictionary

2. Real-word Errors

1. Local syntactic errors: **Their was a problem*
2. Long-distance syntactic errors: **The key to the cabinets are on the table*
3. Semantic errors: *I read the brook*

- Solving this problem is more difficult:
    - The result is still a valid word → not flagged by a dictionary
    - Surrounding **context** must be considered

How common are spelling errors?

- About 2–3% of all typed words on a full-size keyboard are misspelled by proficient adults (Flor et al., 2015)

Table 2. Summary statistics for the ETS Spelling Corpus

	GRE Argument	GRE Issue	TOEFL Independent	TOEFL Integrated	TOTAL
Total essays	750	750	750	750	3,000
Essays without misspellings	60	21	18	21	120
Total Word Count	263,578	336,301	212,930	151,031	963,840
Average Word Count	351	448	284	201	321
Total count of Misspellings	5,935	7,962	7,285	5,230	26,412
Misspellings as % of all words	2.25%	2.37%	3.42%	3.46%	2.74%

Figure 1: Flor et al. (2015), p. 112

How common are spelling errors?

- About 2–3% of all typed words on a full-size keyboard are misspelled by proficient adults (Flor et al., 2015)

Table 2. Summary statistics for the ETS Spelling Corpus

	GRE Argument	GRE Issue	TOEFL Independent	TOEFL Integrated	TOTAL
Total essays	750	750	750	750	3,000
Essays without misspellings	60	21	18	21	120
Total Word Count	263,578	336,301	212,930	151,031	963,840
Average Word Count	351	448	284	201	321
Total count of Misspellings	5,935	7,962	7,285	5,230	26,412
Misspellings as % of all words	2.25%	2.37%	3.42%	3.46%	2.74%

Figure 1: Flor et al. (2015), p. 112

- Most errors are single-character misspellings (edit distance = 1)

How common are spelling errors?

- About 2–3% of all typed words on a full-size keyboard are misspelled by proficient adults (Flor et al., 2015)

Table 2. Summary statistics for the ETS Spelling Corpus

	GRE Argument	GRE Issue	TOEFL Independent	TOEFL Integrated	TOTAL
Total essays	750	750	750	750	3,000
Essays without misspellings	60	21	18	21	120
Total Word Count	263,578	336,301	212,930	151,031	963,840
Average Word Count	351	448	284	201	321
Total count of Misspellings	5,935	7,962	7,285	5,230	26,412
Misspellings as % of all words	2.25%	2.37%	3.42%	3.46%	2.74%

Figure 1: Flor et al. (2015), p. 112

- Most errors are single-character misspellings (edit distance = 1)
- On a mobile phone, however, about 40% of words are misspelled (Grammarly, 2019)

How common are spelling errors?

- About 2–3% of all typed words on a full-size keyboard are misspelled by proficient adults (Flor et al., 2015)

Table 2. Summary statistics for the ETS Spelling Corpus

	GRE Argument	GRE Issue	TOEFL Independent	TOEFL Integrated	TOTAL
Total essays	750	750	750	750	3,000
Essays without misspellings	60	21	18	21	120
Total Word Count	263,578	336,301	212,930	151,031	963,840
Average Word Count	351	448	284	201	321
Total count of Misspellings	5,935	7,962	7,285	5,230	26,412
Misspellings as % of all words	2.25%	2.37%	3.42%	3.46%	2.74%

Figure 1: Flor et al. (2015), p. 112

- Most errors are single-character misspellings (edit distance = 1)
- On a mobile phone, however, about 40% of words are misspelled (Grammarly, 2019)
- More multi-error misspellings and real-word errors due to auto-complete (e.g., ***restaurant*** → typed as ***restuarnt*** → auto-corrected to ***restart***)

Building a simple spell-checker

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits
- Keep only words in the *dictionary* from a *corpus*

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits
- Keep only words in the *dictionary* from a *corpus*
- Pick the most frequent candidate

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits
- Keep only words in the *dictionary* from a *corpus*
- Pick the most frequent candidate
- Example:

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits
- Keep only words in the *dictionary* from a *corpus*
- Pick the most frequent candidate
- Example:
  - Input: *langage*

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits
- Keep only words in the *dictionary* from a *corpus*
- Pick the most frequent candidate
- Example:
  - Input: *langage*
  - Candidates: *language*, *lineage*

Baseline spell checker (Peter Norvig)

- Generate all candidate words within 1–2 edits
- Keep only words in the *dictionary* from a *corpus*
- Pick the most frequent candidate
- Example:
    - Input: *langage*
    - Candidates: *language*, *lineage*
    - Output: **language**

But there’s a problem...

- Some typos are more likely than others

But there's a problem...

- Some typos are more likely than others
    - Adjacent key slips (e.g., *friemd* → *friend*)

But there's a problem...

- Some typos are more likely than others
  - Adjacent key slips (e.g., *friemd* → *friend*)
  - Transpositions (e.g., *teh* → *the*)

But there’s a problem...

- Some typos are more likely than others
  - Adjacent key slips (e.g., *friemd* → *friend*)
  - Transpositions (e.g., *teh* → *the*)
- Baseline only looks at **frequency**, not how errors happen

But there's a problem...

- Some typos are more likely than others
    - Adjacent key slips (e.g., *friemd* → *friend*)
    - Transpositions (e.g., *teh* → *the*)
- Baseline only looks at **frequency**, not how errors happen
- We need a better model: **noisy channel**

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)
- **P(observed|w)** = likelihood of making that typo

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)
- **P(observed|w)** = likelihood of making that typo
- Example:

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)
- **P(observed|w)** = likelihood of making that typo
- Example:
  - Input: *recieve*

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)
- **P(observed|w)** = likelihood of making that typo
- Example:
  - Input: *recieve*
  - Candidates: *recipe*, **receive**

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)
- **P(observed|w)** = likelihood of making that typo
- Example:
    - Input: *recieve*
    - Candidates: *recipe*, **receive**
    - Baseline (frequency only) → *recipe*

Noisy channel spell checker

Formula
$\arg\max_w P(\text{observed} \mid w) \cdot P(w)$

- **P(w)** = prior probability (word frequency)
- **P(observed|w)** = likelihood of making that typo
- Example:
  - Input: *recieve*
  - Candidates: *recipe*, **receive**
  - Baseline (frequency only) → *recipe*
  - Noisy channel (frequency + typo likelihood) → **receive**

Thinking about a more complex spell-checker

Why context matters in spell-checking

Example: Someone types:

```
You put the catt before the horse.
```

- The word `catt` is not found in a dictionary or corpus $\Rightarrow$ likely a misspelling.

Why context matters in spell-checking

Example: Someone types:

```
You put the catt before the horse.
```

- The word `catt` is not found in a dictionary or corpus $\Rightarrow$ likely a misspelling.
- A simple spell-checker (like Norvig's) would:

Why context matters in spell-checking

Example: Someone types:

`You put the catt before the horse.`

- The word `catt` is not found in a dictionary or corpus $\Rightarrow$ likely a misspelling.
- A simple spell-checker (like Norvig's) would:
  - Consider all known words with edit distance 1 from `catt`

Why context matters in spell-checking

Example: Someone types:

`You put the catt before the horse.`

- The word `catt` is not found in a dictionary or corpus ⇒ likely a misspelling.
- A simple spell-checker (like Norvig's) would:
  - Consider all known words with edit distance 1 from `catt`
  - Choose the most frequent candidate: `cat`

Why context matters in spell-checking

Example: Someone types:

`You put the catt before the horse.`

- The word `catt` is not found in a dictionary or corpus $\Rightarrow$ likely a misspelling.
- A simple spell-checker (like Norvig's) would:
  - Consider all known words with edit distance 1 from `catt`
  - Choose the most frequent candidate: `cat`
- But the better correction is actually `cart`, because:

Why context matters in spell-checking

Example: Someone types:

`You put the catt before the horse.`

- The word `catt` is not found in a dictionary or corpus $\Rightarrow$ likely a misspelling.
- A simple spell-checker (like Norvig's) would:
    - Consider all known words with edit distance 1 from `catt`
    - Choose the most frequent candidate: `cat`
- But the better correction is actually `cart`, because:
    - `put the cart before the horse` is a common English expression

Why context matters in spell-checking

Example: Someone types:

`You put the catt before the horse.`

- The word `catt` is not found in a dictionary or corpus ⇒ likely a misspelling.
- A simple spell-checker (like Norvig's) would:
  - Consider all known words with edit distance 1 from `catt`
  - Choose the most frequent candidate: `cat`
- But the better correction is actually `cart`, because:
  - `put the cart before the horse` is a common English expression
  - `put the cat before the horse` is not

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`
- **Bigram** = two-word sequence: `the cat`

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`
- **Bigram** = two-word sequence: `the cat`
- **Trigram** = three-word sequence: `put the cat`

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`
- **Bigram** = two-word sequence: `the cat`
- **Trigram** = three-word sequence: `put the cat`

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`
- **Bigram** = two-word sequence: `the cat`
- **Trigram** = three-word sequence: `put the cat`

How do we use this?

- Count all *n*-grams (e.g., bigrams) in a large corpus

Demo: `https://huggingface.co/spaces/liujch1998/infini-gram`

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`
- **Bigram** = two-word sequence: `the cat`
- **Trigram** = three-word sequence: `put the cat`

How do we use this?

- Count all *n*-grams (e.g., bigrams) in a large corpus
- Use frequency of phrases to estimate how likely a candidate is *in context*

Demo: `https://huggingface.co/spaces/liujch1998/infini-gram`

Using N-grams to model context

N-grams are sequences of *n* elements (e.g., words or characters):

- **Unigram** = one word: `the`
- **Bigram** = two-word sequence: `the cat`
- **Trigram** = three-word sequence: `put the cat`

How do we use this?

- Count all *n*-grams (e.g., bigrams) in a large corpus
- Use frequency of phrases to estimate how likely a candidate is *in context*
- `put the cart` is more frequent than `put the cat`

Demo: `https://huggingface.co/spaces/liujch1998/infini-gram`

Other approaches

- **Statistical Language Models (n-grams)** Use probability of surrounding context e.g., *I went to the shcool* → "school" is more probable

Summary: Traditional = simple but context-blind
Modern = complex but context-aware

Other approaches

- **Statistical Language Models (n-grams)** Use probability of surrounding context e.g., *I went to the shcool* → "school" is more probable
- **Neural Spell Checkers (Deep Learning)** Seq2Seq / Transformer-based models generate corrected text Examples: ChatGPT, Grammarly, Google Docs

Summary: Traditional = simple but context-blind
Modern = complex but context-aware

Other approaches

- **Statistical Language Models (n-grams)** Use probability of surrounding context e.g., *I went to the shcool* → “school” is more probable
- **Neural Spell Checkers (Deep Learning)** Seq2Seq / Transformer-based models generate corrected text Examples: ChatGPT, Grammarly, Google Docs
- **Hybrid Approaches** Combine edit distance with language models; pick the highest probability candidate

Summary: Traditional = simple but context-blind
Modern = complex but context-aware

Real-Word Errors and Grammar

- Not all mistakes are spelling errors → some are **real-word errors**.

Real-Word Errors and Grammar

- Not all mistakes are spelling errors → some are **real-word errors**.
  - Example: *I want to by a book* → “by” is valid, intended: *buy*

Real-Word Errors and Grammar

- Not all mistakes are spelling errors → some are **real-word errors**.
    - Example: *I want to by a book* → “by” is valid, intended: *buy*
- Real-word errors often overlap with **grammar errors**.

Real-Word Errors and Grammar

- Not all mistakes are spelling errors → some are **real-word errors**.
  - Example: *I want to by a book* → "by" is valid, intended: *buy*
- Real-word errors often overlap with **grammar errors**.
  - Example: *Their going to school* → all words exist, but grammar is wrong (*They're*)

Real-Word Errors and Grammar

- Not all mistakes are spelling errors → some are **real-word errors**.
    - Example: *I want to by a book* → "by" is valid, intended: *buy*
- Real-word errors often overlap with **grammar errors**.
    - Example: *Their going to school* → all words exist, but grammar is wrong (*They're*)
- Modern systems therefore blur the line between spell checking and grammar checking, using **context-aware models** to handle both (which we'll talk about in the next class).

Wrap-up

Wrap-up

Key idea: ~~Spelling errors are annoying~~

Wrap-up

Key idea: ~~Spelling errors are annoying~~
Spelling errors vary by types

Wrap-up

Key idea: ~~Spelling errors are annoying~~
Spelling errors vary by types More questions to think about:

- What about the spacing errors?
- What about in other languages that have different encoding systems?