Computational Linguistics

3. Regular Expressions and Edit Distance

Xiaojing Bai

Tsinghua University

https://bxjthu.github.io/CompLing

At the end of this session you will

- understand how a finite state automaton is related to regular expressions and regular languages;
- be able to work with basic regular expressions in pattern matching;
- understand how to quantify the similarity between two strings with Minimum Edit Distance;
- understand the basics of structured programs.

Recap

- An FSA describes a finite set of states together with event-driven transitions between states, with transitions indicated by labelled arcs.
- Possible events were drawn from a finite set called the alphabet.
- There is a start state and several final states.
- The sequence of events that leads from the start state to a final state is said to be a sequence that is accepted by the FSA. (acceptor/recognizer vs. generator)
- The set of all accepted sequences is called a regular language, which can also be defined with a regular expression or a regular grammar.

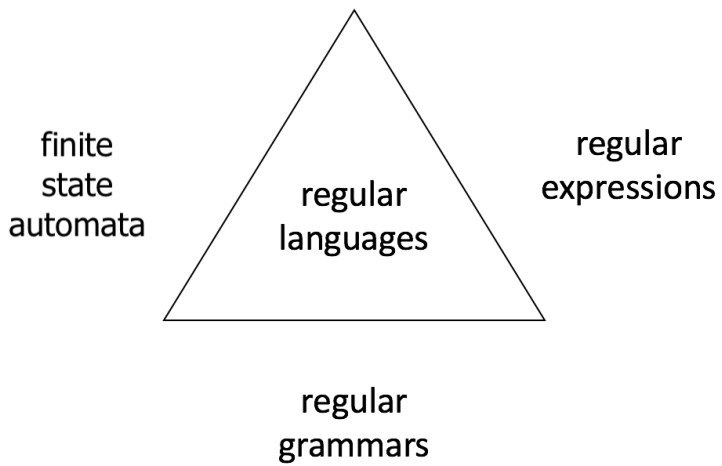
Recap: Yet another formal description of the sheep talk

- The sheep talk

  baa!
  baaa!
  baaaa!
  baaaaa!
  …

- RE for the sheep talk

  /baa+!/



Three equivalent ways of describing regular languages

Regular expressions

One of the **unsung successes** in standardization in computer science

- The most important tool for describing text pattern → **computational model**
- Useful for searching in texts, with a **pattern** to search for and a corpus of **texts** to search through

Regular expressions and Eliza

Eliza: a program which makes natural language conversation between man and computer possible

Weizenbaum, J. (1966). ELIZA - a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1), 36-45.

"Like the Eliza of Pygmalion fame, it can be made to appear even more civilized ..."

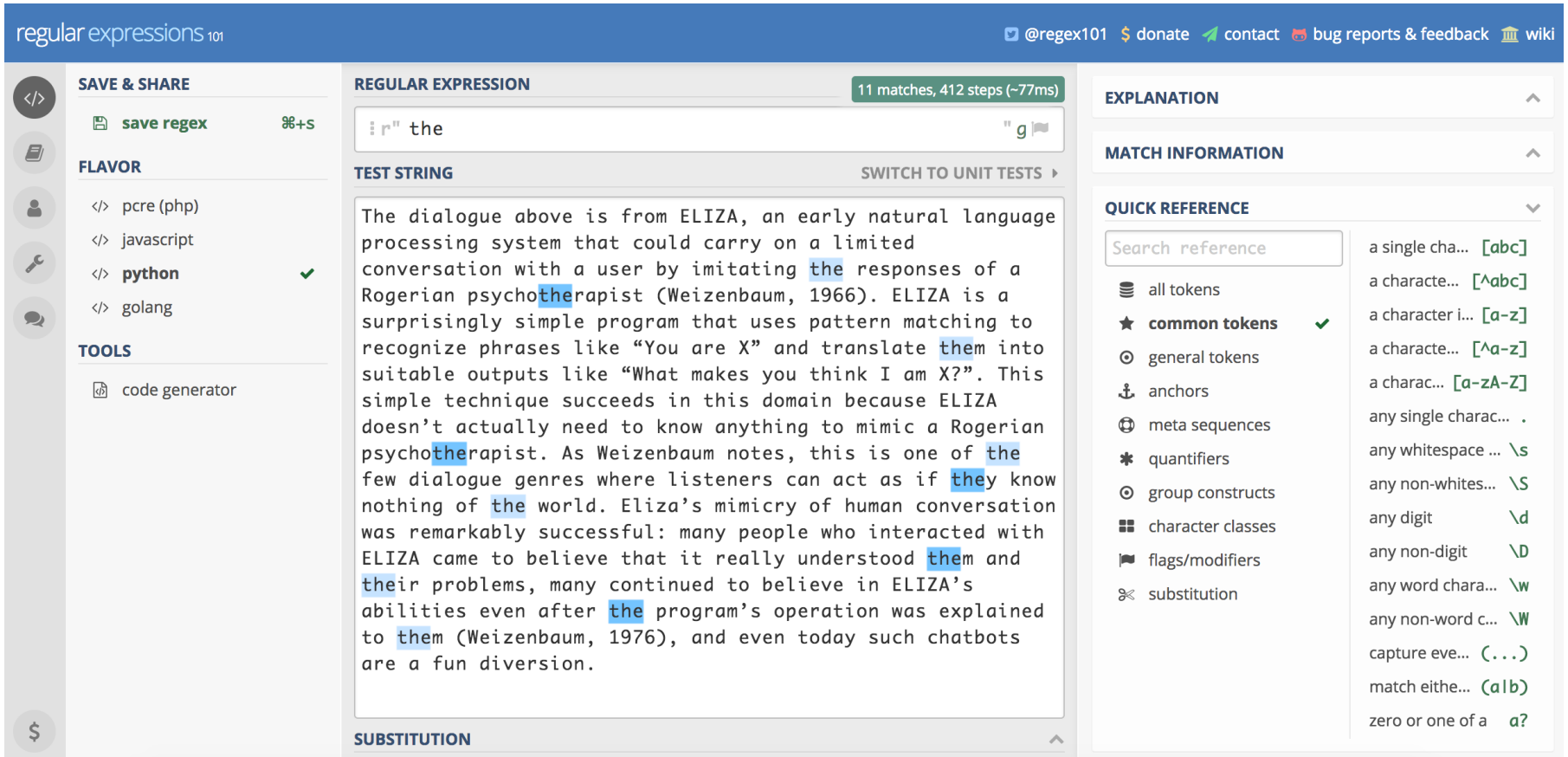
Regular expressions and Eliza

*"...the text is read and inspected for the presence of a **keyword**. If such a word is found, the sentence is transformed according to a **rule** associated with the keyword, if not a content-free remark or, under certain conditions, an earlier transformation is retrieved. The text so computed or retrieved is then printed out."*

E.g. "You are X" → "What makes you think I am X?"

Eliza, the Rogerian Therapist

A quick reference at https://regex101.com/



Processing raw text with regular expressions

3.4 Regular Expressions for Detecting Word Patterns

3.5 Useful Applications of Regular Expressions

To be covered by Quiz 3 (Oct. 17)

Two kinds of errors

- **False positives**: matching strings that we should not have matched (e.g. *there*, *then*, *other*)
- **False negatives**: not matching strings that we should have matched (e.g. *The*)

Two kinds of errors

- **False positives**: matching strings that we should not have matched (e.g. *there, then, other*)
- **False negatives**: not matching strings that we should have matched (e.g. *The*)

Reducing the error rate in NLP applications: two antagonistic efforts

- Increasing **accuracy** or **precision** (minimizing false positives)
- Increasing **coverage** or **recall** (minimizing false negatives).

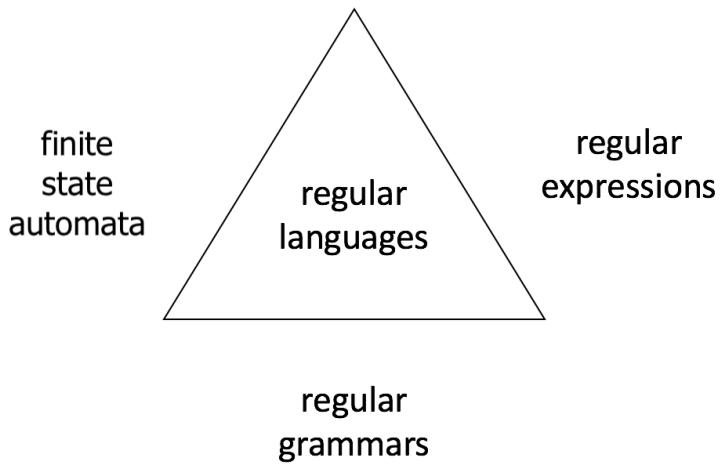
Why are these two efforts "antagonistic"?

Food for your thought

- Sophisticated sequences of regular expressions are often the first model for text processing tasks
- Regular expressions as features in machine learning classifiers
- Any use of RE that you can think of?

RE and FSA

Three equivalent ways of describing regular languages



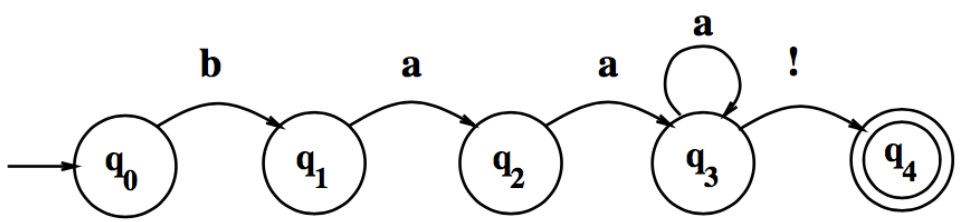
- The Chomsky hierarchy
- Natural language and its complexity
- Formal models and formal languages
- Power of formal models: complexity of the phenomena they can describe

Formal language

- A set of strings, each composed of symbols from a finite symbol-set (alphabet)
- Characterized by a model *m* (such as a particular FSA)

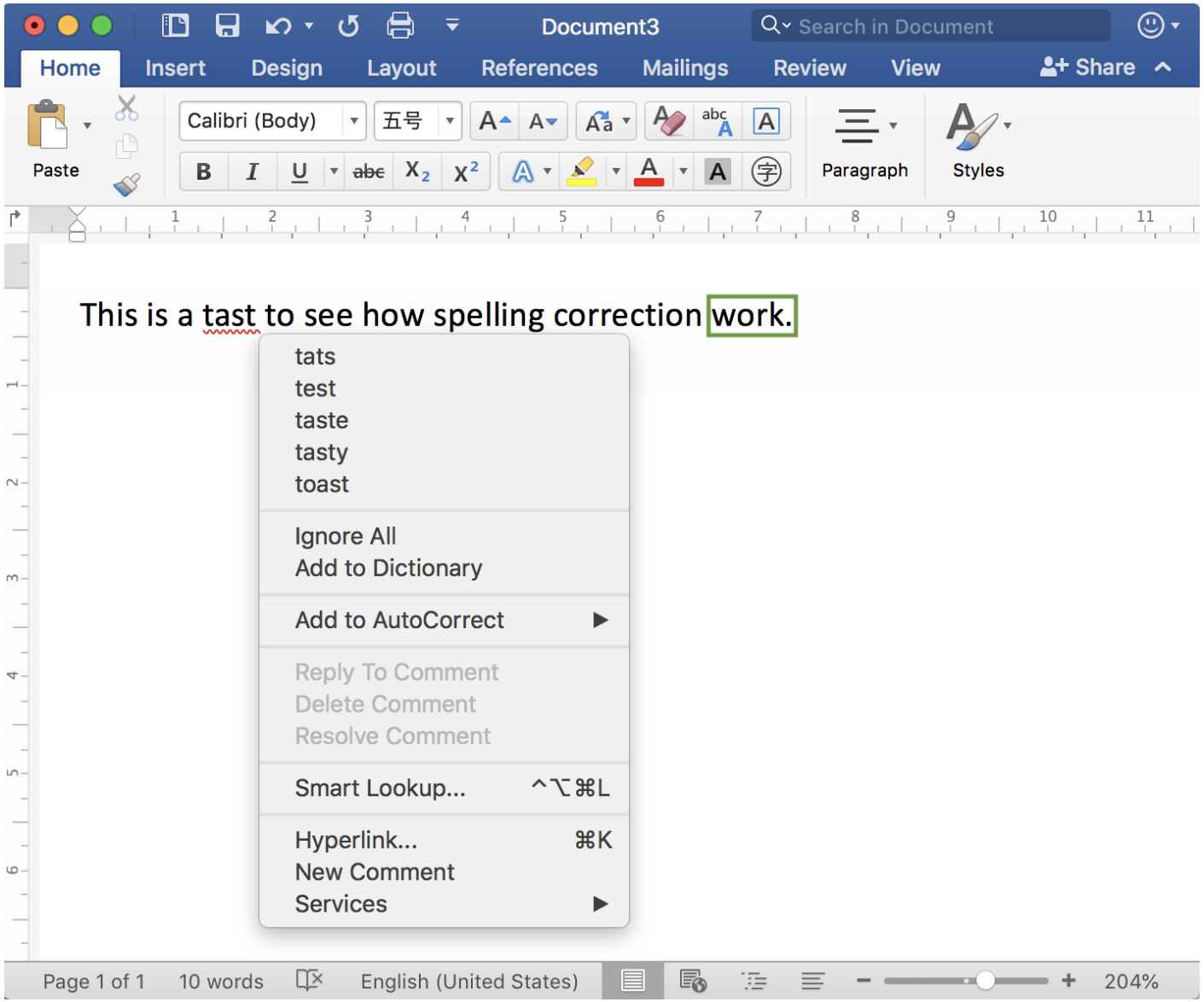
  E.g. *L(m)* = {*baa!, baaa!, baaaa!, baaaaa!, baaaaaa!, …*}

  The sheep talk automaton helps us **recognize** and **generate** the sheeptalk.



  Usefulness: a finite set of symbols to define an infinite set

- Formal language vs. natural language


Document3
Search in Document
Home
Insert
Design
Layout
References
Mailings
Review
View
Share
Paste
Calibri (Body)
五号
Paragraph
Styles
This is a tast to see how spelling correction work.
tats
test
taste
tasty
toast
Ignore All
Add to Dictionary
Add to AutoCorrect
Reply To Comment
Delete Comment
Resolve Comment
Smart Lookup... ^⌥⌘L
Hyperlink... ⌘K
New Comment
Services
Page 1 of 1
10 words
English (United States)
204%


李建 中国好声音 导师 - Google
https://www.google.com/search?hl=zh-CN&ei=8jC8W47KMsLTsAe564rABg&q=李建+中国好声音+导师&oq=李建+中国好声音+导师&gs_l=psy-ab.3...9081.1...
Google
李建 中国好声音 导师
登录
全部
图片
新闻
视频
更多
设置
工具
找到约 4,880,000 条结果 （用时 0.66 秒）
显示的是以下查询字词的结果： 李健 中国好声音 导师
仍然搜索： 李建 中国好声音 导师
如何评价中国好声音李健的表现？ - 知乎
https://www.zhihu.com/answer/447071462
2018年7月20日 - 中国好声音」：我们不听音乐，只看李健。这次「好声音」的导师阵容一改以前的风格，竟然是4男0女。老面孔周杰伦还在，＂场控＂庾澄庆也还在。
《中国好声音》李健被其他3位导师“孤立”？网友：因为他们有的你没有_ ...
https://3g.163.com/idol/article/DMN5VQ0C0517AA86.html
2018年7月14日 - 7月13日，《中国好声音》回归，四位导师分别是周杰伦、李健、谢霆锋和庾澄庆。节目一开始，谢霆锋和庾澄庆弹着吉他以一曲《让我一次爱个够》点燃 ...
视频
13:54
1:24
1:46
《好声音》旦增获得最高分，李健表情很喜悦，宿涵表情震惊
【2018好声音独家导师花絮】李健聊“南薛北张”段子手Sing!China官方超清
【2018好声音独家导师花絮】哈林邀李健“二人转”遭辞演？ Sing!China ...

How similar are two strings?

- Spelling correction

  E.g.
  tast vs. tats | *test* | *taste* | *tasty* | *toast*
  李建 vs. 李健 (More to consider in the case of search engines!)

- Coreference

  E.g.
  Stanford President John Hennessy
  Stanford University President John Hennessy

- Also for Machine Translation, Information Extraction, Speech Recognition

Quantify the similarity between two strings

Minimum Edit Distance:
the minimum number of editing operations needed to transform one into the other

- Insertion (i)
- Deletion (d)
- Substitution (s)

Cost/weight (Levenshtein, 1966)

- If each operation has cost of 1
- If substitutions cost 2

```
I N T E * N T I O N
| | | | | | | | | |
* E X E C U T I O N

d s s   i s
```

Read: The Minimum Edit Distance Algorithm

At the end of this session you will

- understand how a finite state automaton is related to regular expressions and regular languages;
- be able to work with basic regular expressions in pattern matching;
- understand how to quantify the similarity between two strings with Minimum Edit Distance;
- understand the basics of structured programs.

Homework

- Read and practice: (Quiz 3 on Oct. 17, 2018)
  - NLTK Book: 3.4 Regular Expressions for Detecting Word Patterns; 3.5 Useful Applications of Regular Expressions
- Review:
  - J+M_2
  - J+M_second_edition_2 (2.2)
  - J+M_second_edition_3 (3.1-3.7)

Next session

N-gram Language Models