LaSEWeb: Automating Search Strategies over Semi-Structured Web Data

Oleksandr Polozov  
University of Washington  
polozov@cs.washington.edu

Sumit Gulwani  
Microsoft Research  
sumitg@microsoft.com
Motivation: search engine micro-segments

Walter Diemer
Walter E. Diemer was an accountant and inventor of bubble gum. Born and raised in Philadelphia, Pennsylvania, Diemer was working as an accountant at Fleer in 1926 when the company president sought to cut costs by making their own gum base. The company's founder, Frank Henry Fleer, had previously made a batch of bubble gum in 1906...
Motivation: search engine micro-segments
Motivation: search engine micro-segments
Motivation: search engine micro-segments
Repetitive search tasks

Structured databases

- Precise, but limited in content
- No time-sensitive information
- Provide no context (sources)
Repetitive search tasks

Structured databases

- Precise, but limited in content
- No time-sensitive information
- Provide no context (sources)

Web mining scripts

- Two extremes:
  - Powerful ML, which has to be re-learned for each micro-segment
  - Fragile HTML layout parser
- Inaccessible for end-users
LaSEWeb Query Language

• A semantic scripting language for semi-structural information extraction from the Web
• Models natural patterns from the humans’ search strategies

LaSEWeb interpreter

• Explores multiple webpages, clusters different answer candidates, and provides context for each answer
• Makes use of state-of-the-art NLP/ML/PL algorithms
Example: phone number

\[ \vec{v} = ("Sumit Gulwani") \]

\[
\begin{aligned}
\text{let } & \eta_t = \text{Emphasized}(v_1) \text{ in} \\
\text{let } & \eta_b = \text{AttributeLookup}(\text{Syn("phone")}, \ell_a) \text{ in} \\
& \text{Union}(\eta_t, \eta_b) \\
\text{where } & \text{Regex}(\ell_a, "\/(d+)/\W*(d+\W*(d+))") \\
\text{where } & \text{Layout}(\eta_t, \eta_b, \text{Down}) \text{ and } \text{Nearby}(\eta_t, \eta_b)
\end{aligned}
\]
Example: phone number

\[ \vec{v} = ("\text{Sumit Gulwani"}) \]

let \( \eta_t \) = \text{Emphasized}(v_1) \text{ in }
let \( \eta_b \) = \text{AttributeLookup}(\text{Syn("phone"), } \ell_a) \text{ in }
\text{Union}(\eta_t, \eta_b)

where \( \text{Regex}(\ell_a, "/(d+)\W*d + \W*d+" ) \)
where \( \text{Layout}(\eta_t, \eta_b, \text{Down}) \) and \( \text{Nearby}(\eta_t, \eta_b) \)

- Visual attributes
Example: phone number

\[ \vec{v} = ("\text{"Sumit Gulwani""]) \]

\[
\text{let } \eta_t = \text{Emphasized}(v_1) \text{ in }
\text{let } \eta_b = \text{AttributeLookup(Syn("phone"), } \mathcal{L}_a) \text{ in }
\text{Union}(\eta_t, \eta_b)
\]

where \(\text{Regex}(\mathcal{L}_a, "(\d+)\W*\d+\W*\d+\")\)

where \(\text{Layout}(\eta_t, \eta_b, \text{Down}) \text{ and } \text{Nearby}(\eta_t, \eta_b)\)

- Visual attributes
- Implicit table detection
Example: phone number

\[ \vec{v} = ("\text{Sumit Gulwani}"") \]

```
let \( \eta_t \) = Emphasized(\( v_1 \)) in
let \( \eta_b \) = AttributeLookup(Syn("phone"), \( \ell_a \)) in
Union(\( \eta_t, \eta_b \))
where Regex(\( \ell_a \), "(\(d+\))\W*\d + \W*\d+")
where Layout(\( \eta_t, \eta_b \), Down) and Nearby(\( \eta_t, \eta_b \))
```

- Visual attributes
- Implicit table detection
- Linguistic patterns
Example: phone number

\[ \vec{v} = ("Sumit Gulwani") \]

\[
\begin{align*}
\text{let } \eta_t &= \text{Emphasized}(v_1) \text{ in} \\
\text{let } \eta_b &= \text{AttributeLookup}(\text{Syn}(\text{"phone"}), \ell_a) \text{ in} \\
& \quad \text{Union}(\eta_t, \eta_b) \\
\text{where } \text{Regex}(\ell_a, \"(\d+)\W*d+\W*d+\") \\
\text{where } \text{Layout}(\eta_t, \eta_b, \text{Down}) \text{ and } \text{Nearby}(\eta_t, \eta_b)
\end{align*}
\]

- Visual attributes
- Implicit table detection
- Linguistic patterns
- Clustering across webpages
Language Structure

**Visual patterns**
- Match: webpage layout, style, end-user appearance
- Use: in-memory rendering, DOM analysis
- *Nearby, Emphasized, Layout, CSS ...*

**Structural patterns**
- Match: relational patterns on implicit tables
- Use: table detection, plain text analysis using programming-by-example technologies
- *VLOOKUP, AttributeLookup ...*

**Linguistic patterns**
- Match: semantic text properties
- Use: POS tagging, sentence parsing, entity recognition, synonymy detection...
- *Syn, POS, Entity, NP, SameSentence ...*

---

Program interpreter: “user emulation” algorithm
Program interpreter: “user emulation” algorithm

\[ \vec{v} = \text{"computer"} \]
Program interpreter: “user emulation” algorithm

\[ \vec{v} = \text{"computer"} \]

LaSEWeb Engine

LaSEWeb “inventors” MS script

Seed query
Program interpreter: “user emulation” algorithm

\[ \vec{v} = ("computer") \]

LaSEWeb Engine

Seed query

LaSEWeb “inventors” MS script

Q_1: “John Atanasoff”
Q_2: “John Vincent Atanasoff”
Q_3: “Charles Babbage”
Q_4: “Babbage, C.”
Q_5: “konrad zuse”
Program interpreter: “user emulation” algorithm

\[ \vec{v} = \text{"computer"} \]

LaSEWeb Engine

LaSEWeb “inventors” MS script

Seed query

\[ \text{score}(C_i) = \frac{1}{|U|} \sum_{j=1}^{\left|\mathcal{U}\right|} \sum_{s \in C_i} c(s, u_j) \cdot \frac{c(u_j)}{c(u_j)} \]

“John Atanasoff”

“John Vincent Atanasoff”

“Charles Babbage”

“Babbage, C.”

“konrad zuse”
Program interpreter: “user emulation” algorithm

\[ \vec{v} = ("computer") \]

LaSEWeb Engine

Seed query

LaSEWeb "inventors" MS script

John Atanasoff (14.5%)
http://www.computerhope.com
http://www.ehow.com
http://inventors.about.com

Charles Babbage (10.5%)
http://www.buzzle.com
http://www.ask.com

... 

\[ \text{score}(C_i) = \frac{1}{|U|} \sum_{j=1}^{|U|} \sum_{s \in C_i} \frac{c(s, u_j)}{c(u_j)} \]

“John Atanasoff”
“John Vincent Atanasoff”
“Charles Babbage”
“Babbage, C.”
“konrad zuse”

Who invented the computer - Answers.com
 wiki.answers.com | Categories | Technology | Inventions
 Inventor of the Computer Many say the first computer is the "difference engine." The first of those devices was conceived in 1786 by J.H. Müller.

The History of Computers - Computer History Timeline
 inventors.about.com/library/bioindex.htm
 A history of computers and the inventors involved with each computer invention milestone - a timeline with detailed history features.

Computer - Wikipedia, the free encyclopedia
 en.wikipedia.org/wiki/Computer
 A computer is a general purpose device that can be programmed to carry out a set of arithmetic or logical operations automatically. Since a sequence of operations can...
Experiments

- ~95% precision and 71% recall on factoid micro-segments
  - For micro-segments: Precision measured by random sampling, based on top-3 results
  - For end-user repetitive search tasks: Precision/recall measured manually
- Average execution time: ~5 sec/webpage
  - Depends on the rendering settings
- Current setting: offline deployment / database population
Summary & Future work

• Typical patterns of human search strategies in a scripting language for IE
  • Match semi-structured Web content
  • Existing cross-disciplinary technologies used as building blocks
  • Exploit information redundancy across multiple webpages

• Applications:
  1. Micro-segments of factoid questions in search engines
  2. Repeatable batch data extraction tasks for end-users
  3. Structured database population from free Web text
  4. English language comprehension problem generation

• Future work:
  • Automatic query execution plans in the language
  • Integration with “natural language → logic” engines
Summary & Future work

• Typical patterns of human search strategies in a scripting language for IE
• Match semi-structured Web content
• Existing cross-disciplinary technologies used as building blocks
• Exploit information redundancy across multiple webpages

Applications:
1. Micro-segments of factoid questions in search engines
2. Repeatable batch data extraction tasks for end-users
3. Structured database population from free Web text
4. English language comprehension problem generation

• Future work:
  • Automatic query execution plans in the language
  • Integration with “natural language → logic” engines

1. The principal characterized his pupils as _________ because they were pampered and spoiled by their indulgent parents.
2. The commentator characterized the electorate as _________ because it was unpredictable and given to constantly shifting moods.
   (a) cosseted                (b) disingenuous      (c) corrosive             (d) laconic               (e) mercurial
Summary & Future work

• Typical patterns of human search strategies in a scripting language for IE
  • Match semi-structured Web content
  • Existing cross-disciplinary technologies used as building blocks
  • Exploit information redundancy across multiple webpages

• Applications:
  1. Micro-segments of factoid questions in search engines
  2. Repeatable batch data extraction tasks for end-users
  3. Structured database population from free Web text
  4. English language comprehension problem generation

• Future work:
  • Automatic query execution plans in the language
  • Integration with “natural language → logic” engines
Thanks for listening!

Questions?