NLP for the Web / Tools

Yves Petinot

Columbia University

February 4th, 2010
You project will most likely involve one or many of the following components:

- Data acquisition
- Offline Data processing / Labeling
- IR component
- Online Data processing
- Front-end
You project will most likely involve one or many of the following components:

- Data acquisition
- Offline Data processing / Labeling
- IR component
- Online Data processing
- Front-end

What kind of resources can you use?
Available Tools

Generic platforms that may be useful for your projects:

NLTK - http://www.nltk.org/
Python-based corpus readers, tokenizers, stemmers, taggers, parsers, chunkers, etc. comes with various corpora and samples (Brown, PTB, etc.)

GATE - http://gate.ac.uk/
Java-based framework to build NLP pipelines rich library of open source components

Clairlib - http://www.clairlib.org
Perl-based for those of you who took one of Prof. Radev's courses (SET/NET)
Available Tools

Generic platforms that may be useful for your projects:

- **NLTK** - http://www.nltk.org/
  - Python-based
  - corpus readers, tokenizers, stemmers, taggers, parsers, chunkers, etc.
  - comes with various corpora and samples (Brown, PTB, etc.)
Available Tools

Generic platforms that may be useful for your projects:

  - Python-based
  - corpus readers, tokenizers, stemmers, taggers, parsers, chunkers, etc.
  - comes with various corpora and samples (Brown, PTB, etc.)

- **GATE** - [http://gate.ac.uk/](http://gate.ac.uk/)
  - Java-based
  - framework to build NLP pipelines
  - rich library of open source components
Available Tools

Generic platforms that may be useful for your projects:

- **NLTK** - http://www.nltk.org/
  - Python-based
  - corpus readers, tokenizers, stemmers, taggers, parsers, chunkers, etc.
  - comes with various corpora and samples (Brown, PTB, etc.)

- **GATE** - http://gate.ac.uk/
  - Java-based
  - framework to build NLP pipelines
  - rich library of open source components

- **Clairlib** - http://www.clairlib.org
  - Perl-based
  - for those of you who took one of Prof. Radev’s courses (SET/NET)
Data acquisition - A few pointers ...

Your data-set is likely to be - but not necessarily - Web-based
Data acquisition - A few pointers ...

Your data-set is likely to be - but not necessarily - Web-based

- wget/curl + lynx
  - don’t underestimate them, can take you a long way ...
Your data-set is likely to be - but not necessarily - Web-based

- wget/curl + lynx
  - don’t underestimate them, can take you a long way ...
- Nutch for larger scale, intensive crawls
  - http://lucene.apache.org/nutch/
Data acquisition - A few pointers ...

Your data-set is likely to be - but not necessarily - Web-based

- wget/curl + lynx
  - don’t underestimate them, can take you a long way ...
- Nutch for larger scale, intensive crawls
  - http://lucene.apache.org/nutch/
- Search APIs if targeting a particular vertical/set of sites
  - Web Search
  - Site Explorer
  - News Search
Data acquisition - A few pointers ...

Your data-set is likely to be - but not necessarily - Web-based

- wget/curl + lynx
  - don’t underestimate them, can take you a long way ...
- Nutch for larger scale, intensive crawls
  - http://lucene.apache.org/nutch/
- Search APIs if targeting a particular vertical/set of sites
  - Web Search
  - Site Explorer
  - News Search
- Other APIs which maybe relevant to you: del.icio.us API, Twitter API, etc.
NLP Tools

- Many tools available from /proj/nlp/tools
  - stemmers, parsers, NE taggers, etc.
  - code for some of the papers on our reading list
  - make sure you are on compute.cs.columbia.edu, not clic
- Named Entity Taggers & Coreference Resolution (NYU’s ACE)
- Classification / Clustering tools
  - Sentence Clustering
NE Tagging

- Tagging Named Entities (NE) given a plain text
- Example of Tags:
  - **PER:**
    - Individual, Group
  - **ORG:**
    - Sports, Commercial, Media, Governmental, ...
  - **GPE:**
    - Nation (e.g., Russian), Population-Center, ...
  - **TIMEX:**
    - Time and Date (e.g., 2pm, last night, today, ...) 
  - **ENT:**
    - FAC, SUBTYPE="Building-Grounds", (e.g., hospital)
Eddy Arnold (May 15, 1918) is an American country music singer who is second to George Jones in the number of individual hits on the country charts but, according to a formula derived by Joel Whitburn, is the all-time leader in an overall ranking for hits and their time on the charts.
- `<TEXT>`
  - `<ENT ID="7166.txt-1" TYPE="PER" SUBTYPE="Individual">Eddy Arnold</ENT>`
    (  
      `<TIMEX ID="7166.txt-T0" VAL="1918-05-15">May 15, 1918</TIMEX>`
    )
    is  
    - `<ENT ID="7166.txt-2" TYPE="PER" SUBTYPE="Individual">an`  
      - `<ENT ID="7166.txt-3" TYPE="GPE" SUBTYPE="Nation">American</ENT>`
      country music singer  
    </ENT>`
  who is second to  
  - `<ENT ID="7166.txt-4" TYPE="PER" SUBTYPE="Individual">George Jones</ENT>`
    in the number of individual hits on the country charts but, according to a formula derived by  
  - `<ENT ID="7166.txt-5" TYPE="PER" SUBTYPE="Individual">Joel Whitburn</ENT>`
    , is  
  - `<ENT ID="7166.txt-5" TYPE="PER" SUBTYPE="Individual">the all-time leader</ENT>`
    in an overall ranking for hits and their time on the charts.  
`</TEXT>`
- <TEXT>
  <ENT ID="797.txt -1" TYPE="PER" SUBTYPE="Individual ">The vice president</ENT> made
  <ENT ID="797.txt -1" TYPE="PER" SUBTYPE="Individual ">his</ENT> comments as
  <ENT ID="797.txt -1" TYPE="PER" SUBTYPE="Individual ">he</ENT> divided
  <ENT ID="797.txt -1" TYPE="PER" SUBTYPE="Individual ">his</ENT> day between debate preparation and campaigning in
  <ENT ID="797.txt -8" TYPE="GPE" SUBTYPE="State-or-Province ">an important state</ENT> where
  <ENT ID="797.txt -1" TYPE="PER" SUBTYPE="Individual ">he</ENT> is threatening
  <ENT ID="797.txt -9" TYPE="PER" SUBTYPE="Individual ">Republican George W. Bush</ENT> 's expectations of victory.
</TEXT>
NYU’s 2005 ACE system - How to run it ... 

1. source /proj/gale-safe/system/distill/bin/init.sh
NYU’s 2005 ACE system - How to run it ...

1. source /proj/gale-safe/system/distill/bin/init.sh
2. cd /proj/gale-safe/users/sergey/jet
NYU’s 2005 ACE system - How to run it ...

1. source /proj/gale-safe/system/distill/bin/init.sh
2. cd /proj/gale-safe/users/sergey/jet
3. java Xmx500M -cp jet-all-*.jar AceJet.Ace props/MEace06.properties
   **input_files.list location_of_sgm_files/ path_output_ace/**

Where,
- **input_files.list** contains a list of file names with .sgm extension:
- each .sgm file should follow the format:
  
  `<TEXT >your text </TEXT >`

- **location_of_sgm_files** is an absolute path to the location of the sgm files
  (/ at the end) (e.g., /home/ypetinot/sgm/)
- **path_output_ace** is path of the output files.

4. python /proj/gale-safe/users/sergey/scripts/insert_ace_annotations.py
   **input file1.sgm path output ace/input file1.sgm.apf 6**
   > final output for file1.xml
NYU’s 2005 ACE system - How to run it ...

1. source /proj/gale-safe/system/distill/bin/init.sh
2. cd /proj/gale-safe/users/sergey/jet
3. java Xmx500M -cp jet-all-*-*.jar AceJet.Ace props/MEace06.properties
   input_files.list location_of_sgm_files/ path_output_ace/
   Where,
   - input_files.list contains a list of file names with .sgm extension:
   - each .sgm file should follow the format:
     
     <TEXT >your text </TEXT >

   - location_of_sgm_files is an absolute path to the location of the sgm files (/ at the end) (e.g., /home/ypetinot/sgm/)
   - path_output_ace is path of the output files.

4. python /proj/gale-safe/users/sergey/scripts/insert_ace_annotations.py
   input_file1.sgm path_output_ace/input_file1.sgm.apf 6
   >final_output_for_file1.xml
Sentence Clustering

- single-link hierarchical clustering, based on similarity threshold
- source /proj/nlp/tools/cluster_sentences/init.sh
- /proj/nlp/tools/cluster_sentences/runcluster.sh input.txt

input.txt

S1##1
S2##2
...
SN##N
Sentence Clustering

- single-link hierarchical clustering, based on similarity threshold
- source /proj/nlp/tools/cluster_sentences/init.sh
- /proj/nlp/tools/cluster_sentences/runcluster.sh input.txt

---

**input.txt**

Jackson, who was born in 1972, is a good man##1
He studied at Columbia University ##2
He was born in 1962##3
Yes, I agree with you##4
Sentence Clustering

- single-link hierarchical clustering, based on similarity threshold
- source /proj/nlp/tools/cluster_sentences/init.sh
- /proj/nlp/tools/cluster_sentences/runcluster.sh input.txt

Output

Jackson, who was born in 1972, is a good man
He was born in 1962
He studied at Columbia University
Yes, I agree with you
IR Component

Lucene
- open source, industry standard
- customizable
  - http://lucene.apache.org/java/docs/

Indri - Information Retrieval Engine
- More research oriented, more flexible for you to tinker with
- Relevance feedback, etc.
- Rich query language.
  - For example:
    - #syn( #1(united states) #1(united states of america) )
    - #2(white house) – matches "white X house" (where X is any word or null)
- More details: http://ciir.cs.umass.edu/metzler/indriquerylang.html
IR Component

- Lucene
  - open source, industry standard
  - customizable
  - http://lucene.apache.org/java/docs/
IR Component

- Lucene
  - open source, industry standard
  - customizable
  - http://lucene.apache.org/java/docs/

- Indri - Information Retrieval Engine
  - More research oriented, more flexible for you to tinker with
  - Relevance feedback, etc.
  - Rich query language.
  
  For example:
  
  #syn( #1(united states) #1(united states of america) )
  #2(white house) – matches ”white X house” (where X is any word or null)

  More details: http://ciir.cs.umass.edu/ metzler/indriquerylang.html
Running Indri on compute ...

- source /proj/nlp/tools/cluster_sentences/init.sh
- cd /proj/gale-safe/system/distill/bin
- python ./postIndri.py -r "Clinton" >doc_ids.xml → list of document ids
- /usr/bin/python ./postIndri.py -p '#combine(...)' >doc_ids.xml
- python ./readUmass.py -o oqua <doc_ids.xml >output.xml
Recommendations ...

- Most of the tools you might need can be found in `/proj/nlp/tools`
  - make sure you are on `compute.cs.columbia.edu`, *not clic*
- The rest is freely available on the Web
- Use these tools wisely:
  - should allow you to focus on core components of your project
  - you don’t have to commit to a single language/framework
  - use scripts to glue components together!
- feel free to ask if you’re facing technical (or non-technical) issues!