CS 5604 spring 2015

Named Entity Recognition

Instructor: Dr. Edward fox

Presenters: Qianzhou du, Xuan Zhang

04/30/2015

Virginia tech, Blacksburg, VA

Table of contents

- Introduction
- Theory
- Implementation
- Parallelization
- Conclusion

NER Concept

- **Named-entity recognition** (NER) is a subtask of **Information Extraction** that seeks to locate and classify elements in text into pre-defined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.

  [Wikipedia]

- Why NER?
  - Event Extraction
  - Question Answering
  - Text summarization

Named Entity Recognition Example

A man opposed to the joint South Korea-U.S. military drills attacked the American ambassador, Mark Lippert, in Seoul Thursday morning.

🟥	Location
🟩	Organization
🟦	Person
🟪	Date

Table of contents

- Introduction
- Theory
- Implementation
- Parallelization
- Conclusion

Problem Definition

- Input: a word sequence
  - word_sequence = $<X_1, X_2, X_3, X_4, \ldots, X_n>$
- Output: their Named-Entity tag sequence
  - tag_sequence = $<Y_1, Y_2, Y_3, Y_4, \ldots, Y_n>$
- Items in $<Y_1, Y_2, Y_3, Y_4, \ldots, Y_n>$ might be person, location, organization, etc..

How to assign the NE tag for a particular word?

- Let us consider the following scenarios:
  - I love the city of New York. (New York is a location)
  - New York Times discloses the inside story. (New York is a news organization)
  - Jeremy Lin unexpectedly led a winning turnaround with New York in 2012. (New York is a sport organization)
- The context is a very important factor to assign the NE tag.

Linear-Chain CRF

- CRF: Conditional Random Field that is an undirected graph whose nodes correspond to YUX. This graph is parameterized in the same way as an Markov network, as a set of factors $\phi_1(D_1), \dots, \phi_m(D_m)$

$$P(Y|X) = \frac{1}{Z(X)} P^{\sim}(Y,X) \qquad P^{\sim}(Y,X) = \prod_{i=1}^{m} \phi_i(D_i) \qquad Z(X) = \sum_{Y} P^{\sim}(Y,X)$$

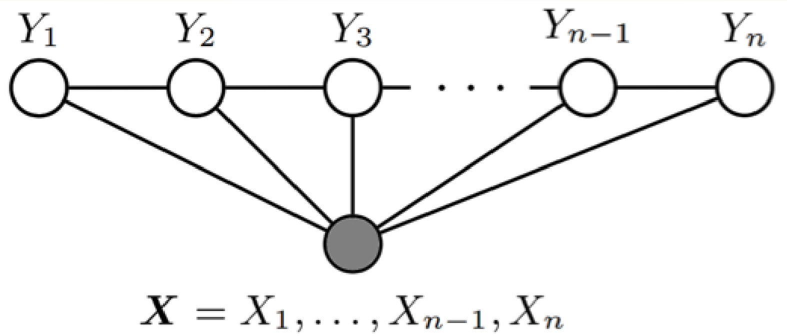
- Goal: Get the conditional distribution P(Y | X), where Y is a set of target variables and X is a set of observed variables.
- Linear-Chain CRF:
  - Based on the basic CRF, and it has only two factors for each word:

$$\phi_t^1(Y_t, Y_{t+1})$$

$$\phi_t^2(Y_t, X_1, \dots, X_T)$$

Linear-Chain CRF

- Graphical Model of Linear-Chain CRF

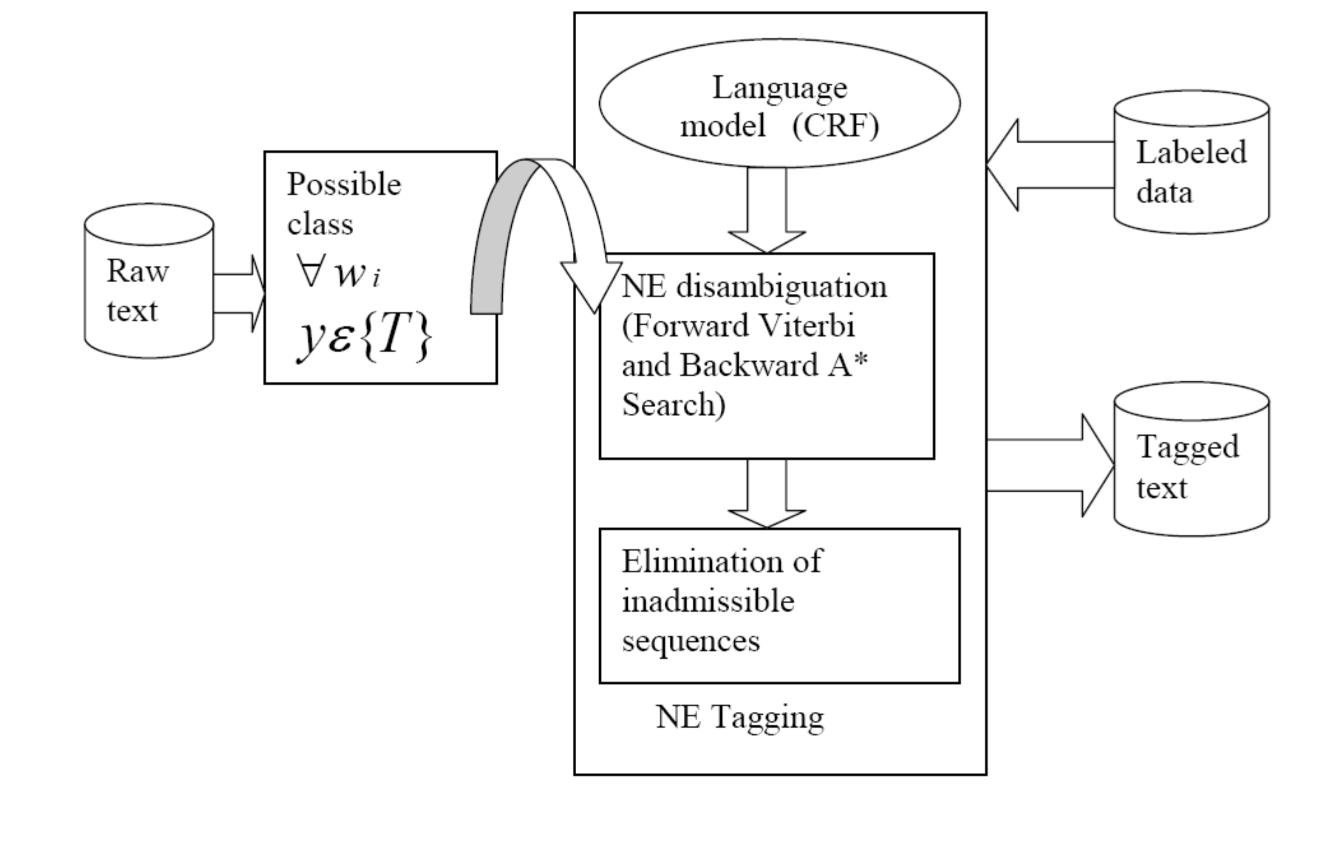


- Two Factors:
  - $\phi_t^1(Y_t, Y_{t+1})$ represents the dependency between neighboring target variables. And $\phi_t^2(Y_t, X_1, \ldots, X_T)$ represents the dependency between a target and its context in the word sequence.
  - Arbitrary features of the entire input word sequence.
  - Log-linear model, but not table factor
- Forward-backward Algorithm to compute the probability distribution
- Viterbi (Dynamic Programming) to choose the best tag sequence by maximizing the probability

Table of contents

- Introduction
- Theory
- Implementation
- Parallelization
- Conclusion

NER Architecture

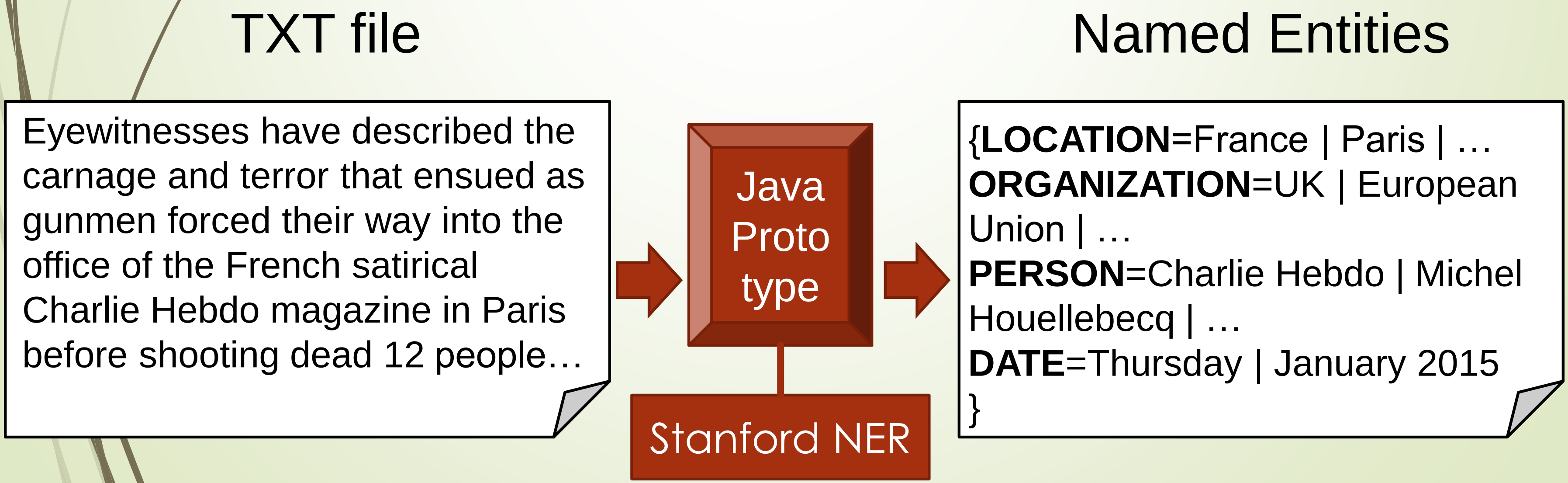


NER System based on CRF

[Asif Ekbal]

NER Tools & Prototype

Tools	Models
Stanford NER	Linear-chain CRF
Illinois Named Entity Tagger	HMM, Neural Network
Alias-i LingPipe	HMM, CRF



TXT file

Eyewitnesses have described the carnage and terror that ensued as gunmen forced their way into the office of the French satirical Charlie Hebdo magazine in Paris before shooting dead 12 people…

Java Proto type

Stanford NER

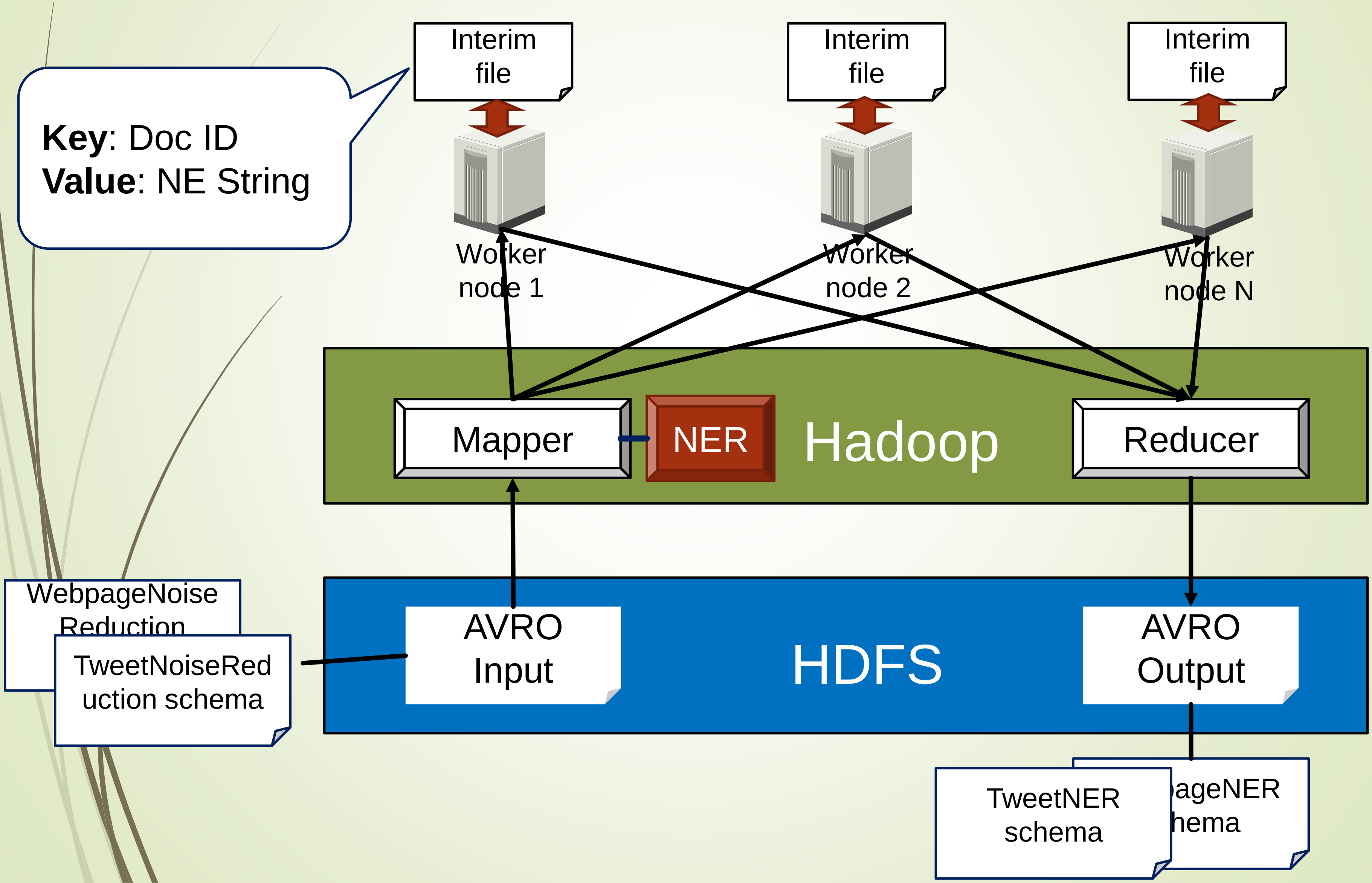
Named Entities

{**LOCATION**=France | Paris | …
ORGANIZATION=UK | European Union | …
PERSON=Charlie Hebdo | Michel Houellebecq | …
DATE=Thursday | January 2015
}

Table of contents

- Introduction
- Theory
- Implementation
- Parallelization
- Conclusion

NER Parallelization on Hadoop



Map-Reduce Implementation

- Driver for Map-Reduce job
- Set configuration

```
Public class NERDriver extends
Configured implements Tool{

  public int run(String[] args) {}

   public static void
   main(String[] args) {}

}
```

- Read a document
- Perform NER
- Write NE string

```
public static class
AvroNERMapper extends
Mapper<AvroKey<WebpageN
oiseReduction>, NullWritable,
Text, Text> {

  protected void
map(AvroKey<WebpageNoise
Reduction> key, NullWritable
value, Context context){}
}
```

- Parse NE string
- Write AVRO file

```
public static class
AvroNERReducer extends
Reducer<Text, Text,
AvroKey<WebpageNER>,
NullWritable> {

   protected void reduce(Text
key, Iterable<Text> value,
Context context){}

}
```

Input & Output

AVRO Schema of Input File (By Noise Reduction Team)

```
{"type": "record", "namespace":
"cs5604.tweet.NoiseReduction", "name":
"TweetNoiseReduction", "fields":
…
}
```

AVRO Schema of Output File (By Hadoop Team)

```
{"namespace": "cs5604.tweet.NER",
"type": "record",
"name": "TweetNER",
"fields": [
   {"name": "doc_id", "type": "string"},
   {"doc": "analysis", "name": "ner_people", "type":
["string", "null"]},
   {"doc": "analysis", "name": "ner_locations",
"type": ["string", "null"]},
   {"doc": "analysis", "name": "ner_dates", "type":
["string", "null"]},
   {"doc": "analysis", "name": "ner_organizations",
"type": ["string", "null"]}
]
}
```

Input & Output (Cont.)

AVRO Output File with Named Entities

```
{u'ner_dates': u'December 08 | December 11', u'ner_locations': None, u'doc_id': u'winter_storm_S--
100052', u'ner_people': None, u'ner_organizations': u'NWS'}

{u'ner_dates': None, u'ner_locations': None, u'doc_id': u'winter_storm_S--10025', u'ner_people':
u'Blaine Countys', u'ner_organizations': None}

{u'ner_dates': None, u'ner_locations': None, u'doc_id': u'winter_storm_S--100229', u'ner_people':
None, u'ner_organizations': u'ALERT Winter Storm Watch'}

{u'ner_dates': None, u'ner_locations': None, u'doc_id': u'winter_storm_S--100364', u'ner_people':
None, u'ner_organizations': u'Heavy Snow Possible Winter Storm Watch | Northeast PA | Coal
Region Endless Mtns'}

...
```

(From winter_storm_S Tweet collection)

Statistics

Collections	Size	Time
winter_storm_S (Tweet)	166 MB	6 Min
storm_B (Tweet)	6.3 GB	10 Min
winter_storm_S (Webpage)	62 MB	4 Min
storm_B (Webpage)	N/A	N/A

Table of contents

- Introduction
- Theory
- Implementation
- Parallelization
- Conclusion

Conclusion

- Investigate the theory of NER
- Implement NER prototype based on the Stanford NER tool
- Parallelize NER on Hadoop
- Export NEs to AVRO files

Acknowledgement

- This project is supported by the US National Science Foundation through grant **IIS - 1319578**.
- We express our deep appreciation to the Instructor Dr. Fox, GTA Sunshin Lee and GRA Mohammed for their help.

Thank You!

Q & A

NER Data/Back-Offs

- ❑ CoNLL-2002 and CoNLL-2003 (British newswire)
  - ▪ Multiple languages: Spanish, Dutch, English, German
  - ▪ 4 entities: Person, Location, Organization, Misc
- ❑ MUC-6 and MUC-7 (American newswire)
  - ▪ 7 entities: Person, Location, Organization, Time, Date, Percent, Money
- ❑ ACE
  - ▪ 5 entities: Location, Organization, Person, FAC, GPE
- ❑ BBN (Penn Treebank)
  - ▪ 22 entities: Animal, Cardinal, Date, Disease, …