NERC

Named Entity Recognition and Classification

Outline

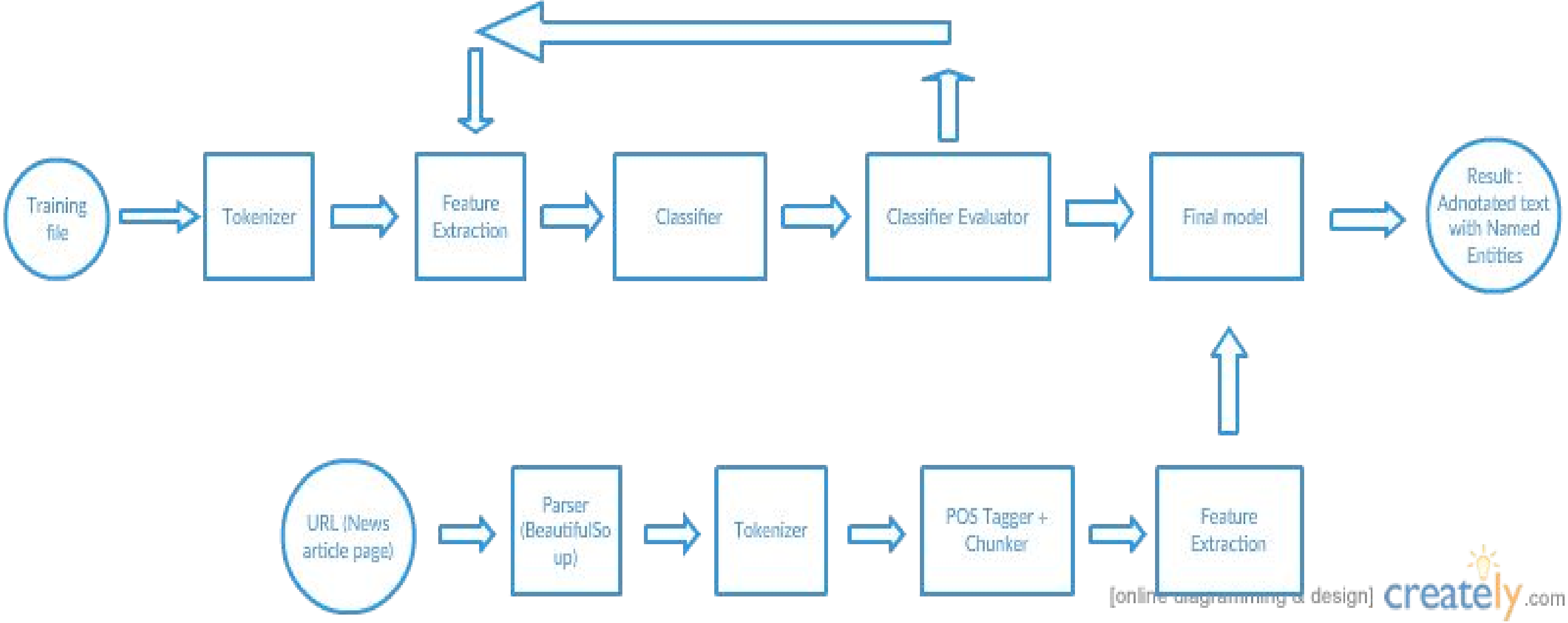
- Overall architecture
- Tools
- Feature extraction
- Classifiers and evaluation
- Preliminary results
- QA

Overall architecture(1)

- Training data
  - **CoNLL 2003**
- Tokenizer
  - (**word**, **POS Tag**, **Chunk Tag**, **NE Tag**)
- POS Tagger and Chunker
  - nltk
- Classification and Evaluation

```
The DT I-NP O
European NNP I-NP I-ORG
Commission NNP I-NP I-ORG
said VBD I-VP O
```

Overall architecture(2)



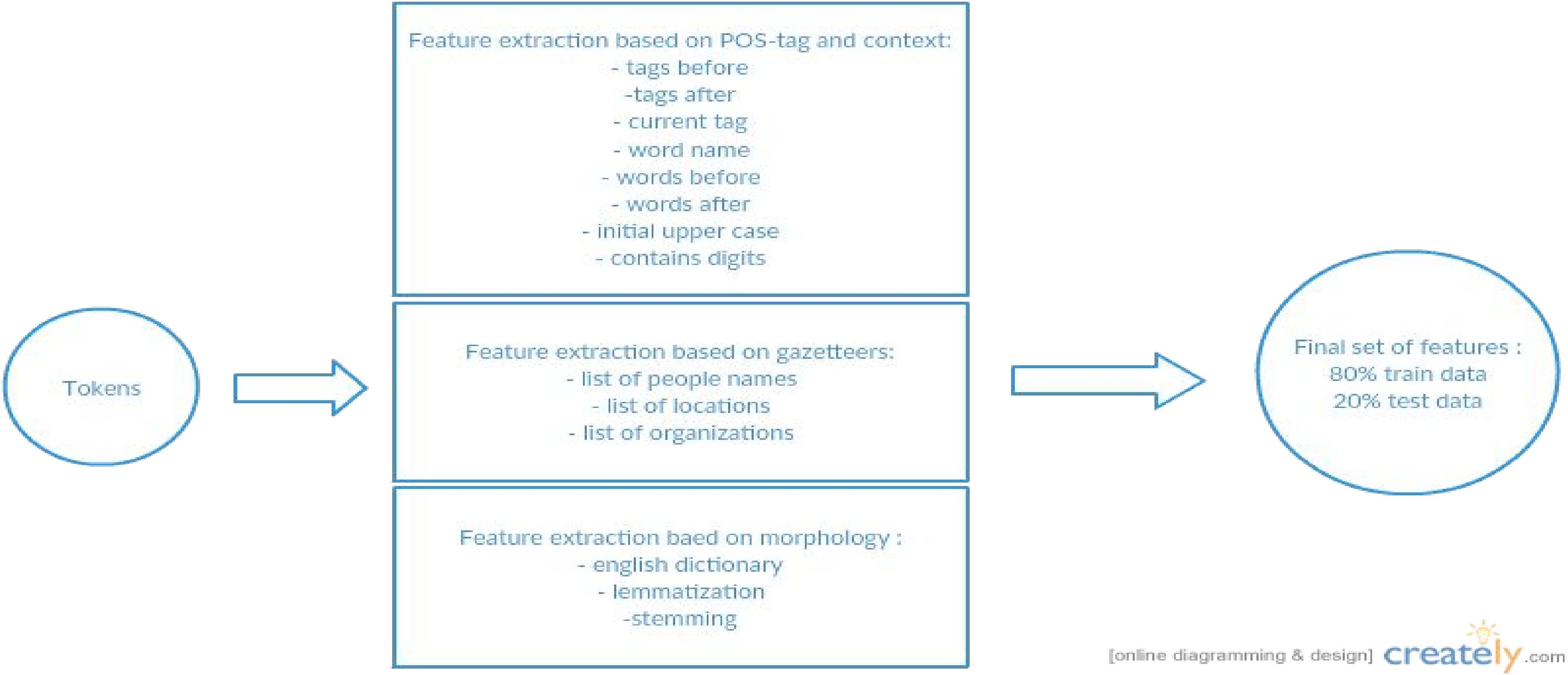
Tools

- Scikit-learn for the Maximum Entropy Classifier (Logistic Regresssion)
- Pandas library for a user-friendly representation of the training and test data
- NLTK library for python
  - POS-Tagger and Chunker
  - Naive Bayes + other classifier
  - POS-Tagged corpuses

Feature extraction(1)

- Feature extraction based on POS-tags and context
  - Current word
  - Previous words
  - Next words
  - POS Tags
  - Word suffixes
- Feature extraction based on gazetteers:
  - Elite Classic → Elite Classic → HTL → Asia/Dubai
- Feature extraction based on morphology
  - WordNet lemmatization

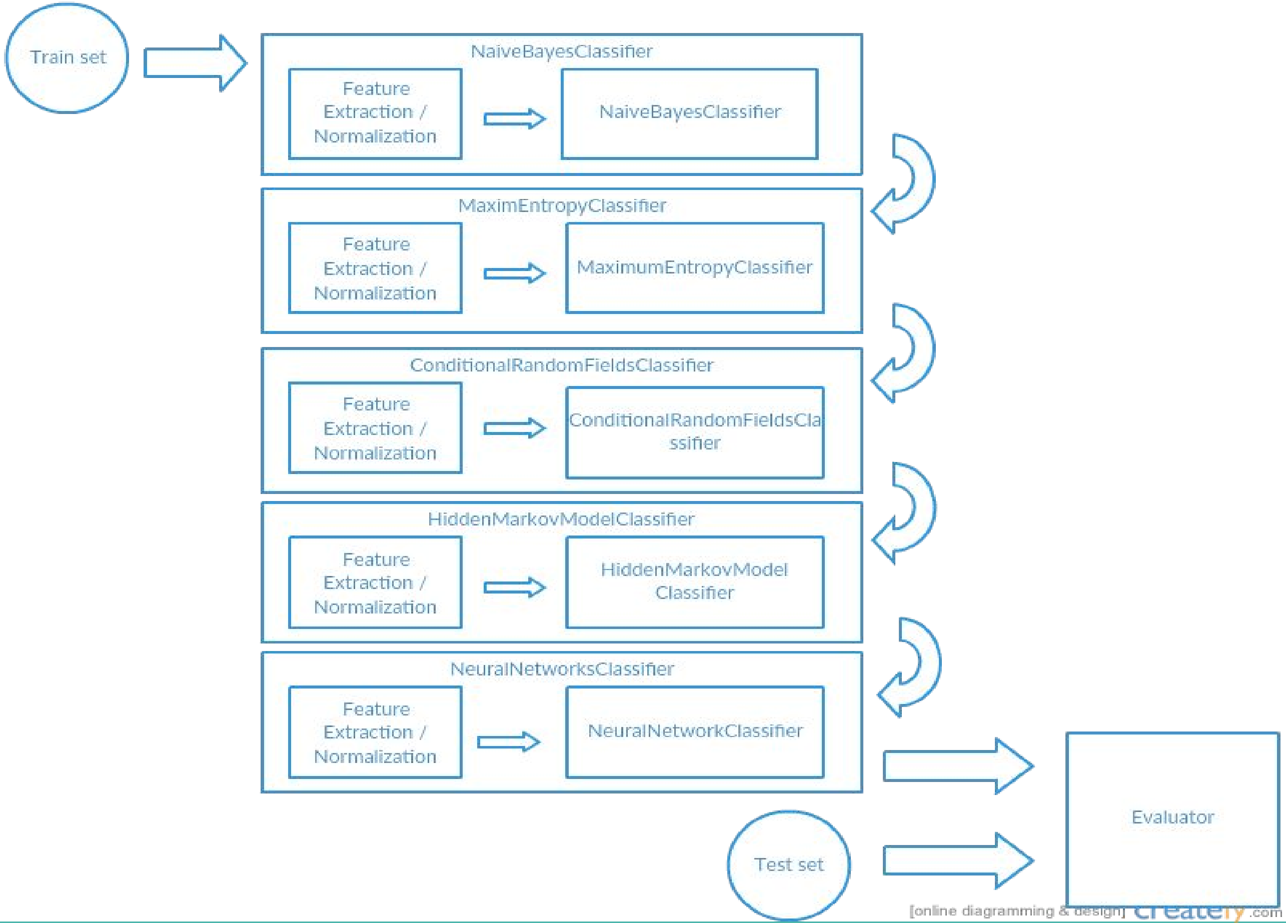
Feature extraction(2)



Classifier and evaluation(1)

- Naive Bayes classifier
- Maximum Entropy Classifier
  - Logistic Regression → scikit-learn
- Neural Network Classifier
  - Word2Vec(Python) + NN(Torch)

Classifier and evaluation(2)



Preliminary results(1) → Naive Bayes

Precision Person: 0.64632 | Recall Person: 0.81137 | Accuracy Person: 0.97018 | **F-score Person: 0.71950**

Precision ORG: 0.49823 | Recall ORG: 0.55752 | Accuracy ORG: 0.95602 | **F-score ORG: 0.52621**

Precision LOC: 0.54844 | Recall LOC: 0.64562 | Accuracy LOC: 0.96427 | **F-score LOC: 0.59308**

Preliminary results(2) → Maximum Entropy

- **CoNLL 2003** corpus
- Logistic Regression → **scikit-learn**
- Comparison with nltk.
- My results:
  - Including “O”-tags:
    - **F1-Score**: 82 % **Precision**: 82.4 % **Recall**: 82.4 %
  - Without “O”-tags:
    - **F1-Score**: 16 % **Precision**: 16.8 %**Recall**: 16.8 %

Preliminary results(3) ⟶Word2Vec + NN (I)

- Dataset sample (ConLL 2003)

Word	POS tag	syntactic chunk Tag	named entity tag
EU	NNP	I-NP	I-ORG
rejects	VBZ	I-VP	O
German	JJ	I-NP	I-MISC
call	NN	I-NP	O
to	TO	I-VP	O
boycott	VB	I-VP	O
British	JJ	I-NP	I-MISC
lamb	NN	I-NP	O
.	.	O	O

Preliminary results(3) →Word2Vec + NN (II)

Word2Vec steps

- process text

  e.g: EU rejects German call to boycott British

- give input to word2vec and get the vectors
  - Set window skipping at 1 and minimum frequency at 0 to catch all the words

- serialize vectors to be used at training NN

Preliminary results(3) →Word2Vec + NN (III)

- Check similarities

Test similarity

```
In [8]: indexes, metrics = model.analogy(pos=['of'], neg=[], n=10)
```

```
In [9]: model.generate_response(indexes, metrics).tolist()
```

```
Out[9]: [(u'from', 0.995997284001497),
 (u'for', 0.995595312942011),
 (u'at', 0.9955636284163529),
 (u'with', 0.9935470112344263),
 (u'by', 0.9933905557391695),
 (u'over', 0.990679873191216),
 (u'in', 0.9901027223658493),
 (u'new', 0.9893901859142927),
 (u'after', 0.9893870905141411),
 (u'bodies', 0.9884139010302534)]
```

Preliminary results(3) →Word2Vec + NN (IV)

```
nn.Sequential {

[input -> (1) -> (2) -> (3) -> (4) -> (5) -> output]

    (1): nn.Linear(80 -> 400)

    (2): nn.ReLU

    (3): nn.Linear(400 -> 800)

    (4): nn.ReLU

    (5): nn.Linear(800 -> 8)

}
```

Results:

- Accuracy: 63%
- Why 63%? Predicting all values as others
- Why predicting all values as others?
- Dataset is unbalanced

1 : 1218
2 : 4
3 : 13959
4 : 3772
5 : 617
6 : 5
7 : 2192
8 : 3

//egal din toate

Preliminary results(3) →Word2Vec + NN (V)

ConfusionMatrix:

[[4	0	964	0	0	0	0	0]		0.413%	[class: 1]
[0	0	4	0	0	0	0	0]		0.000%	[class: 2]
[41	0	11001	1	0	0	0	0]		99.620%	[class: 3] Others
[11	0	3070	0	0	0	0	0]		0.000%	[class: 4]
[1	0	494	0	0	0	0	0]		0.000%	[class: 5]
[0	0	4	0	0	0	0	0]		0.000%	[class: 6]
[9	0	1809	0	0	0	0	0]		0.000%	[class: 7]
[0	0	3	0	0	0	0	0]]		0.000%	[class: 8]

References

[1] Bender, Oliver, Franz Josef Och, and Hermann Ney. "Maximum entropy models for named entity recognition." *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003.
[2] Curran, James R., and Stephen Clark. "Language independent NER using a maximum entropy tagger." *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003.
[3] Chieu, Hai Leong, and Hwee Tou Ng. "Named entity recognition: a maximum entropy approach using global information." *Proceedings of the 19th international conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 2002.
[4] Liu, 2011 "*Web Data Mining*" p.109 - p.120 Ch 3.8 : Support Vector Machines
[5] Bishop,2006 "*Pattern recognition and machine learning*" p.325 - p.357, Ch. 7 : Sparse Kernel Machines
[6] Asif Ekbal and Sivaji Bandyopadhyay - "*Named Entity Recognition using Support Vector Machine: A Language Independent Approach*"
[7] Fredrick Edward Kitoogo and Venansius Baryamureeba - "*A Methodology for Feature Selection in Named Entity Recognition*"
[8] Joel Mickelin - "*Named Entity Recognition with Support Vector Machines*", Master of Science Thesis Stockholm, Sweden 2013
[9] Tomas Mikolov et al. - "Distributed Representations of Words and Phrases and their Compositionality"
[10] Xiang Zhang, Yann LeCun - "Text Understanding from Scratch"

Questions ?

Thank you

for your attention!