Natural Language Processing with Python

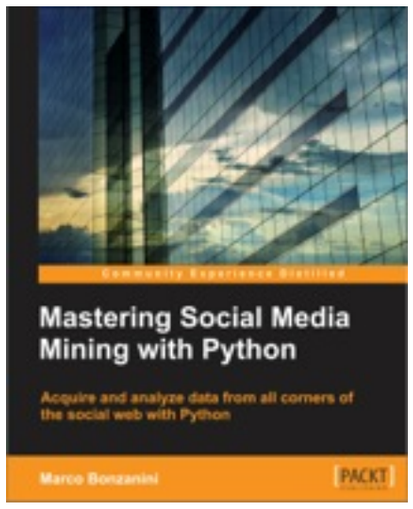
github.com/bonzanini/nlp-tutorial

@MarcoBonzanini and @MiguelMAlvarez

Nice to Meet You

Marco Bonzanini
Freelance Data Scientist

Miguel Martinez-Alvarez
Head of Research



PyData London

SIGNAL

github.com/bonzanini/nlp-tutorial

Schedule

- Intro & Logistics (10m)
- Environment Set Up (10m)
- Exploring Text Data (1h + 15m QA)
- Break (10:45 — 11:15)
- Text Classification (1h)
- Bonus Content (30m + 15m QA)

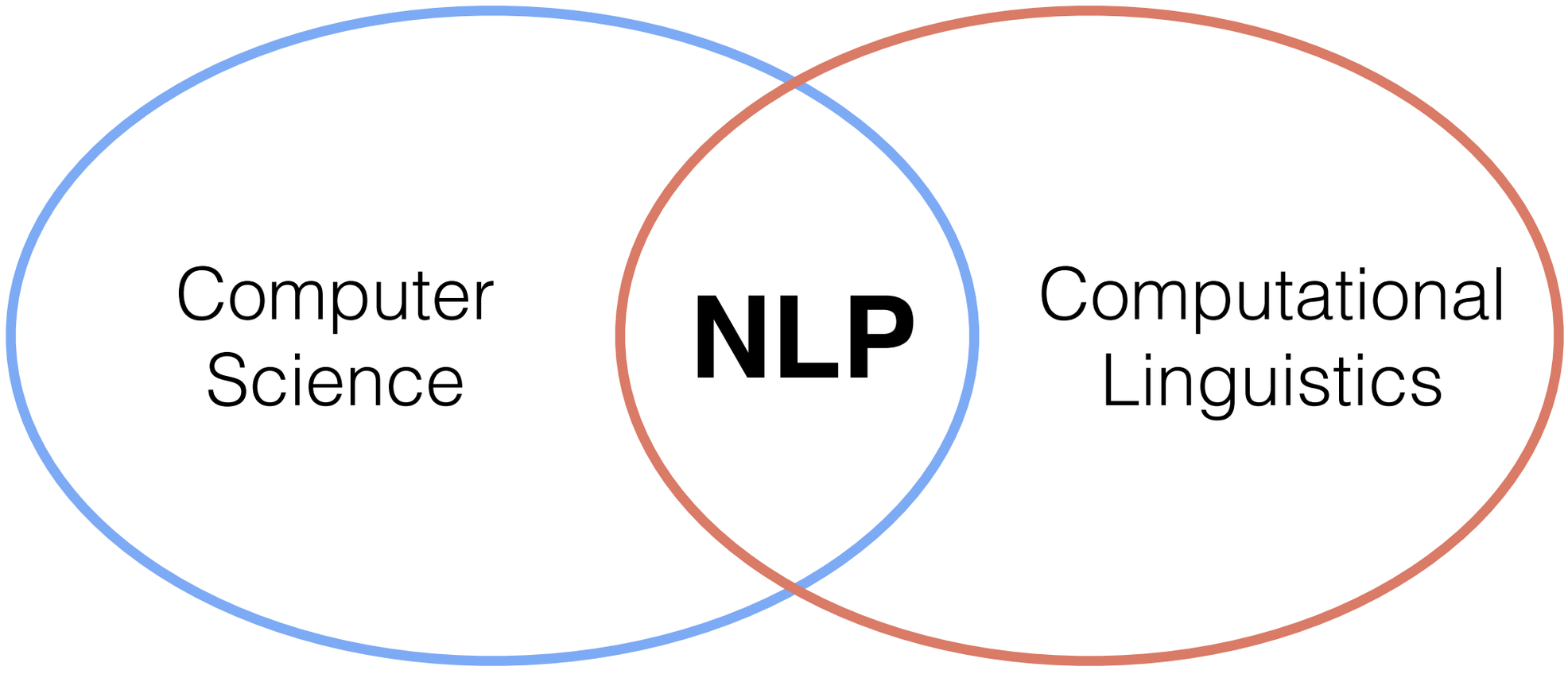
`github.com/bonzanini/nlp-tutorial`

The Audience (You!)

- Know some Python already?
- Know some NLP already?
- Both / None of the above?

`github.com/bonzanini/nlp-tutorial`

Natural Language Processing



github.com/bonzanini/nlp-tutorial

NLP Goals

Text Data  Useful Information
Actionable Insights

`github.com/bonzanini/nlp-tutorial`

Formal vs Natural

```
SELECT name, address
FROM businesses
WHERE business_type = 'pub'
AND postcode area = 'CF10'
```

vs

Where is the nearest pub?

github.com/bonzanini/nlp-tutorial

NLP Applications

- Text Classification
- Text Clustering
- Text Summarisation
- Machine Translation
- Semantic Search
- Sentiment Analysis
- Question Answering
- Information Extraction

`github.com/bonzanini/nlp-tutorial`

Environment Set Up

- Tested with Python 3.4 and 3.5
- Clone the repository:

```
git clone https://github.com/bonzanini/nlp-tutorial
cd nlp-tutorial
```

Environment Set Up (cont'd)

- Set up virtual environment:

```
virtualenv nlp-venv
source nlp-venv/bin/activate
pip install -r requirements.txt
```

Environment Set Up (cont'd)

- Set up virtual environment (alternative):

```
conda create --name nlp-venv python=3.5
source activate nlp-venv
pip install -r requirements.txt
```

Environment Set Up (cont'd)

- Download NLTK data:

```
python -m nltk.downloader \
    punkt stopwords reuters
```

Environment Set Up (cont'd)

- Start up Jupyter notebook:

```
jupyter notebook
```

Exploring Text Data

Goal: Answering Important Questions

What are the most important ingredients in Italian cuisine?

recipes_exploratory_analysis.ipynb

Recipe Analysis: Summary

- Tokenisation
- Counting words
- Stop-words
- Normalisation
- Stemming
- n-grams

pyconuk_exporatory_analysis.ipynb

PyConUK Analysis Summary

- “This talk will …”
- TF-IDF
- We’re going to use scikit-learn

Break

Text Classification

Text Classification

- *“Text categorization (a.k.a. text classification) is the task of assigning predefined categories to free-text documents. It can provide conceptual views of document collections and has important applications in the real world”*

Scholarpedia (Yiming Yang and Thorsten Joachims)

Text Classification

- **Binary**: Only two categories which are mutually exclusive
    - Spam detection, Anomaly detection, Fraud detection, …
- **Multi-class**: Multiple categories, mutually exclusive
    - Language detection, …
- **Multi-label**: Multiple categories with the possibility of multiple (or none) assignments.
    - News Categorisation, Marketing profiling, …

text_classification_Generic.ipynb

Text Classification

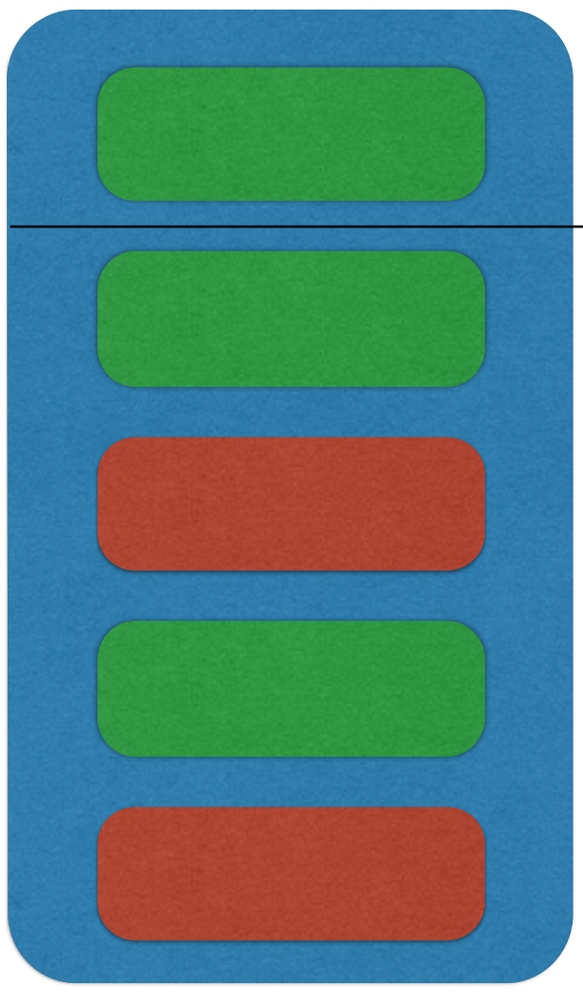
Evaluation

Text Classification Evaluation

- *"If you cannot measure it, you cannot improve it".*
  Lord Kelvin

- Main metrics for **Text** Classification: Precision and Recall

Text Classification Evaluation

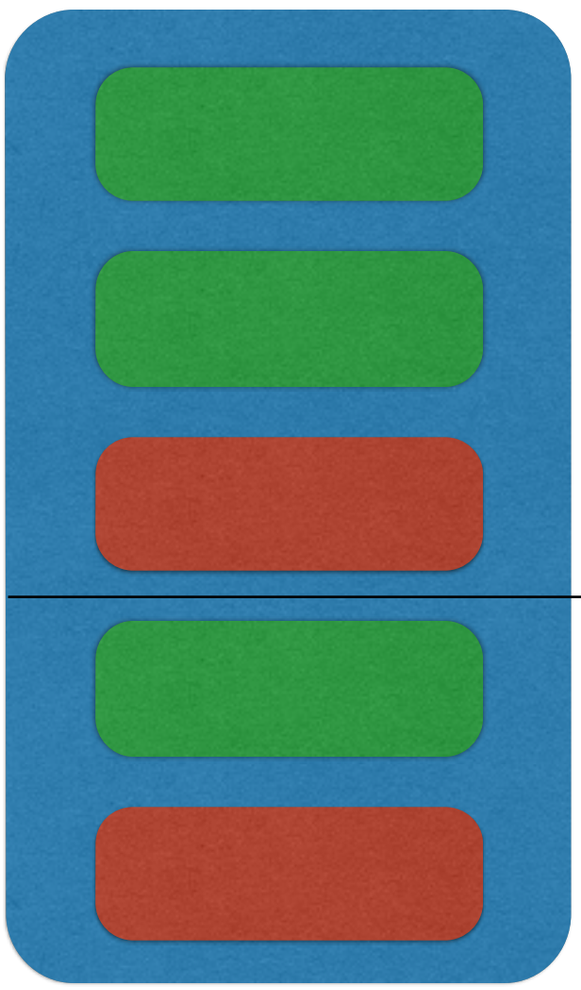


- 1 correct case labelled in the class out of 1 prediction

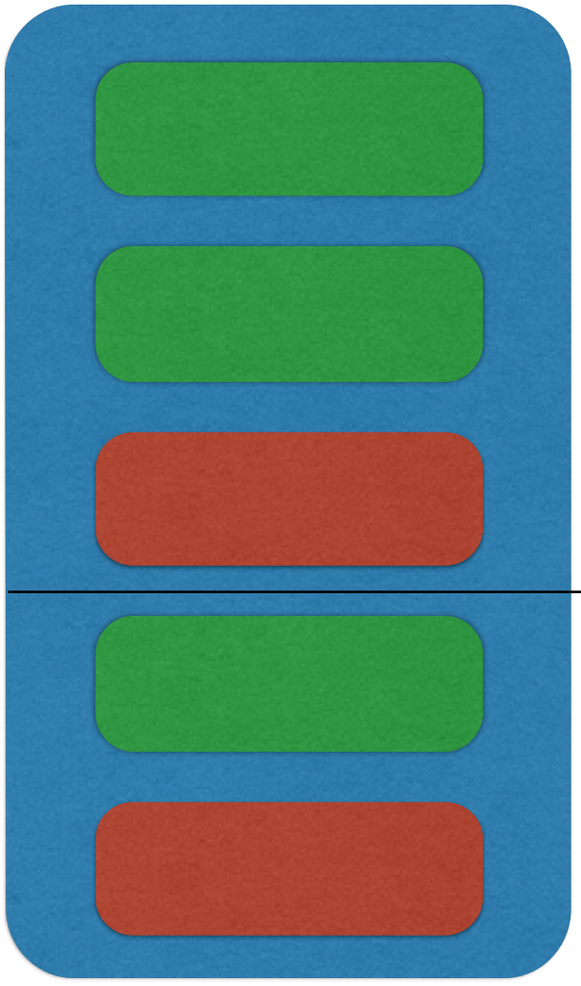
- 1 correct case labelled out of 3 being correct

- **Precision: 100%**
  Recall: 33%

Text Classification Evaluation



Text Classification Evaluation



- 2 correct cases labelled in the class out of 3 predictions

- 2 correct cases labelled out of 3 being correct

- **Precision: 66%**
  Recall: 66%

text_classification_Evaluation.ipynb

Classifying a real collection

`text_classification_Reuters.ipynb`

text_classification_Reuters.ipynb

Text Classification Summary

- Types of Classification Problems
- Document Representations: Vectorizers
- Training and predicting
- Evaluation: Precision *vs* Recall

Questions?