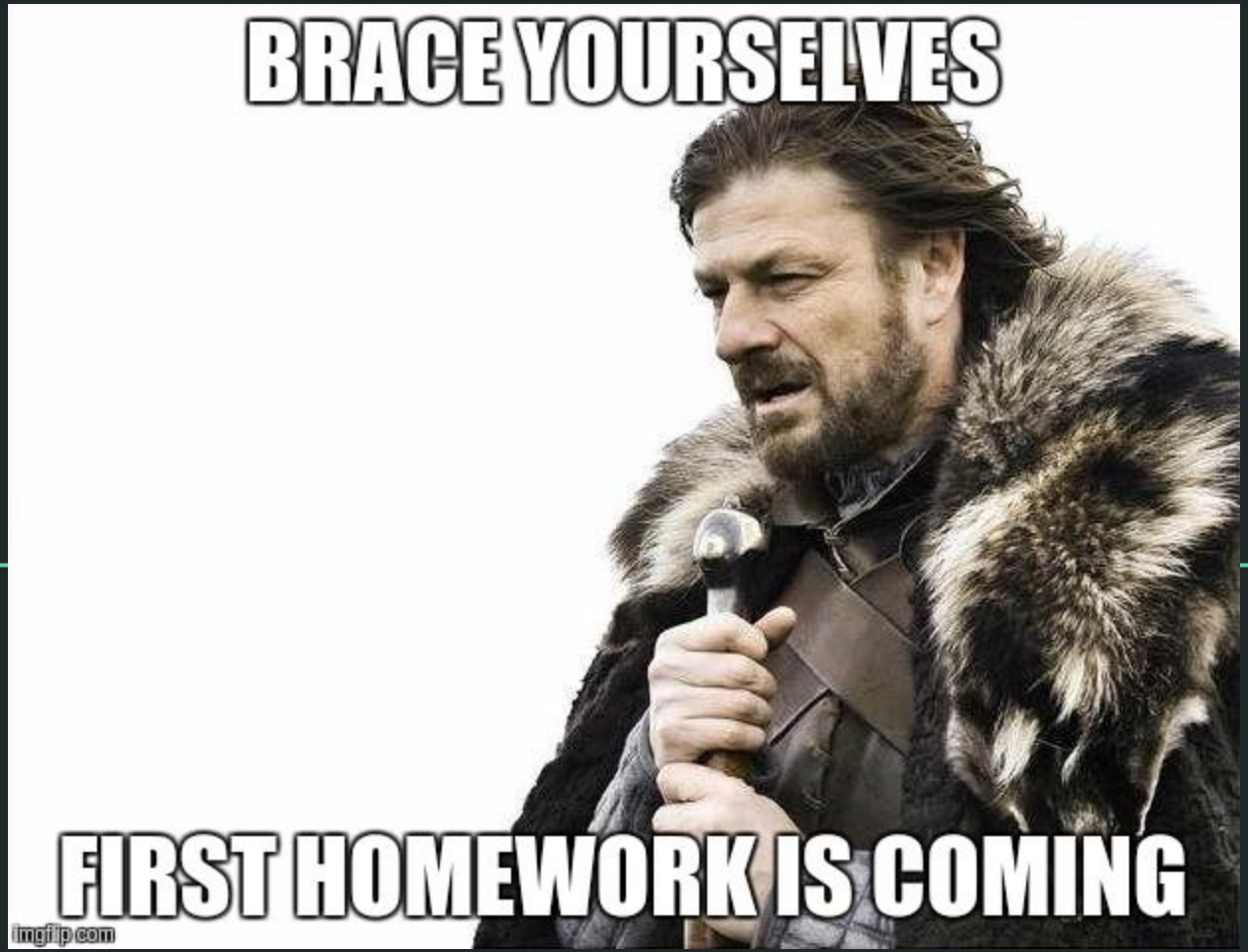

BRACE YOURSELVES
FIRST HOMEWORK IS COMING
imgflip.com

Homework 1

Tommaso Pasini & Valentina Pyatkin

(pasini | pyatkin)@di.uniroma1.it

Supervised Morphological Segmentation

The Model

- Ruokolainen Teemu, et al. **Supervised Morphological Segmentation in a Low-Resource Learning Setting using Conditional Random Fields**. CoNLL. 2013.

- **Idea:** Treat morphological segmentation as a classification problem of a sequence of letters.

- **Tagset:** START, B, M, E, S, STOP

- Each letter in a word is tagged with a tag from the tagset.

CRF with Perceptron

- Conditional Random Field:

$$p(\mathbf{y}|\mathbf{x};\mathbf{w}) \propto \prod_{t=2}^{T} exp(\mathbf{w}^T \mathbf{f}(y_{t-1}, y_t, \mathbf{x}, t))$$

- We learn the parameters (**w**) with the **structured/multiclass perceptron learning algorithm**:
  - for **n** numbers of iterations:
    - for each training instance **x** and its label **y** in the dataset
      - predict $y^* = argmax_{y'} p(y|x;w)$
      - *if* $y = y^*$ do nothing
      - *else* update **w:** $w := w + r[f(y_{t-1}, y_t, x, t) - f^*(y_{t-1}, y_t^*, x, t)]$
      - **r** = learning rate constant, **f** = features for true label, **f***=features for predicted label

Amazing features and how to represent them.

- Every word represents a separate sequence, e.g.: <w>working</w>
- Every letter in that word has its own feature set.
- Hyperparameter: δ ➜ max. length of each feature in the feature set.
- E.g. if δ = 4, the feature set for the letter '**r**' in '<w>wo**r**king</w>' would be:
  - left = {o, wo, <w>w}, right = {r, rk, rki, rkin}
- The feature function is a binary indicator function, meaning the features will be represented as 0s or 1s inside a data structure
  - featureset_**r**= { 'left_o' : 1 ; 'right_rki' : 1 ; 'right_rkin' : 1, ..., 'left_b' : 0}

Machine Learning Recalls

- **Training set:**
  Set of pair (example, label) used for the training phase.
- **Dev set:**
  Set of pair (example, label) used to tune some parameters of the system.
- **Test set**:
  Set of (example, label) used to test the system trained on the training set.

- **Mantra: Training, development and test CANNOT share any (example, label) pair.**

Your Task

Your Task - Overview

- Train your own morphological segmentation model for **English**, as described in **Supervised Morphological Segmentation in a Low-Resource Learning Setting using Conditional Random Fields.**
- You can work **alone** or in **groups of 2**.

Your Task - Overview cont'd

- Lonely workers: Have to complete Task #1.
- Groups: Have to complete Task #1 and Task #2.
- **Use:**
  - **sklearn_crfsuite (python)** https://goo.gl/um56oT
  - **MALLET (java)** https://goo.gl/sCgLLQ

Task #1 - Input Data

- **The Folder we will provide:**
  - homework_1
    - task1_data/
      - training.eng.txt
      - dev.eng.txt
      - test.eng.txt
- **Training, dev and test data** from morpho challenge 2010 (http://morpho.aalto.fi/events/morphochallenge2010), a challenge in the research community to make a fair comparison between morphological systems.
- **Entry format**:
  word <TAB> morpheme_1:lemma_1:tag_1 morpheme_2: ...
  adversaries <TAB> advers:adverse_A ari:ary_s es:+PL

Task #1 - Delivery

- nome_cognome_matricola
    - -src/
    - - task_1/
        - - model/
            - - crf_model.model
    - - homework1_report.pdf

Your Task - How to structure the report

Report has to be 1 page long or 2 if you are a group.

- **Section 1: System overview:**
  How did you implement the model, which language you choose, which library, how did you encode the features etc.

- **Section 2: Results and Analysis:**
  Report the results obtained by your trained model in terms of precision, recall and F1, test different split of the training data and show how (and if) performances increase as number of training examples increases. Report also results with different δ and how you tune it.

- **Section 3: Task_2 (groups only):**
  Report which language you choose for the new training set and results your system scores on your test set.

Task #1 - Delivery cont’d

How to save your model:

- **Python:**

```
import pickle
crf = sklearn_crfsuite.CRF( ... )
pickle.dump(crf, open("output_file", "wb"))
```

- **Java:**

```
CRF classifier = new CRF( ... )
ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream (outputFile));
oos.writeObject (classifier);
oos.close();
```

Task #2 - Input Data

You have to create your own input data in your mother tongue!!!

- **Create a new training set** (of >=100 distinct entries from a Wikipedia page of your choice) in your mother language (e.g., Italian for Italians).
- **Create a new development set** of 25 distinct entries.
- **Create a test set** of 25 distinct entries.

Task #2 - Delivery

- **Your newly created annotations** in the “task_2/” subfolder as specified on the next slide.

- Discuss on the **report the performance of your system** trained, tuned and tested on your annotations.

- **Add the names of your group members to the report.**

Task #2 - Delivery cont’d

- nome_cognome_matricola
    - src/
    - task_1/
        - model/
            - crf_model.model
    - task_2/
        - model/
            - <lang>_model.model
        - data/
            - training.<lang>.txt
            - dev.<lang>.txt
            - test.<lang>.txt
    - homework1_report.pdf

Your Task - How to evaluate

We follow Kurimo et al. 2005 which introduced boundaries evaluation method.

Given all tokenizations produced: (**system output**)

- “cup bearer s’” (BME BMMMME S)
- “boule vard” (BMMME BMME)

And the correct tokenization: (**gold standard**)

- “cup bear er s ’ ” (BME BMME BE S S)
- “boulevard” (BMMMMMMME)

Your Task - How to evaluate - Precision

1. Compute ***H*** = number of correctly placed boundaries (2 between *cup* and *bear* and between *er* and *s* - *count the **E** and **S** in the labels which are at the same index in both segmentations*-)

2. Compute ***I*** = number of incorrect boundaries (1 between *boule* and *vard* - *count the **E** and **S** which are misplaced in the produced tokenization* -)

3. Compute $\text{Precision} = \frac{H}{H + I}$

Your Task - How to evaluate - Recall

1. Compute **H** as defined in the previous slide

2. Compute **D** = the missing boundaries (between *bear* and *er* and between *s* and the apostrophe ‘ - *count the **E** and **S** which are present in the gold tokenization and are not in the system output -)*

3. Compute $\text{Recall} = \frac{H}{H + D}$

Your Task - How to evaluate - F1-score

It is the harmonic mean between Precision and Recall.

$$\text{F1-Score} = \frac{2PR}{(P+R)}$$

Extra Points

- Annotate at least 30 English words with their morphological segmentation.
- Annotate on Google spreadsheet at this address: https://goo.gl/HPbrXg
- Try to avoid duplicates and follow the format shown in Sheet1 of the Google doc.

Extra Points cont’d

- Add the whole annotations sheet to the training data.
- Train your model using the combined training set.
- Add your model (naming it extra_model.model) in the “extra_point/” folder of your submission.

Prizes!!

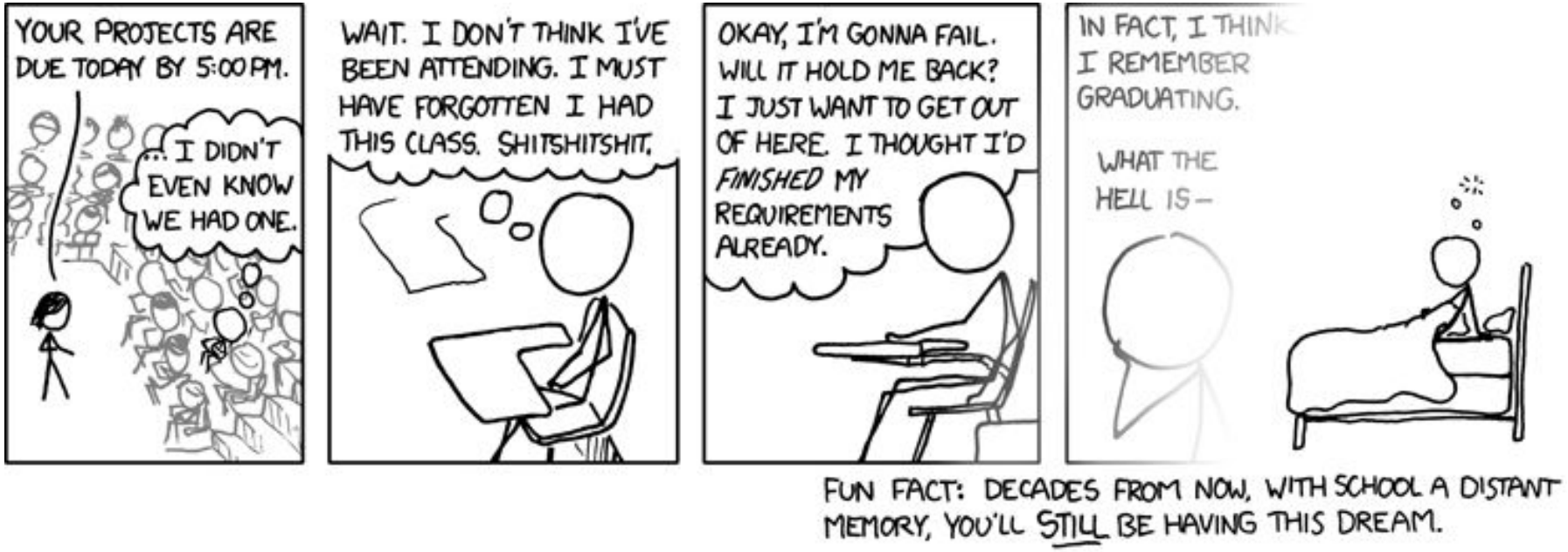
We will publish a leaderboard of the 5 best models (according to their f1 measure) which will win an amazing BabelNet t-shirt!

Our tips

- Read and understand the paper.
- Use python sklearn_crfsuite for their implementation of a CRF or Java MALLET Framework.
- Start with small amount of data from the training set.

The deadline:

- Sunday, 26th of March extra point deadline.
- Sunday, 2nd of April
- Time: 23:59
- http://robertonavigli.com/nlp2017/



Good luck...

Source: https://xkcd.com/557/