THE UNIVERSITY of EDINBURGH

Text Technologies for Data Science

INFR11145

Text Classification (2)

Instructor:
Walid Magdy

07-Nov-2017

Lecture Objectives

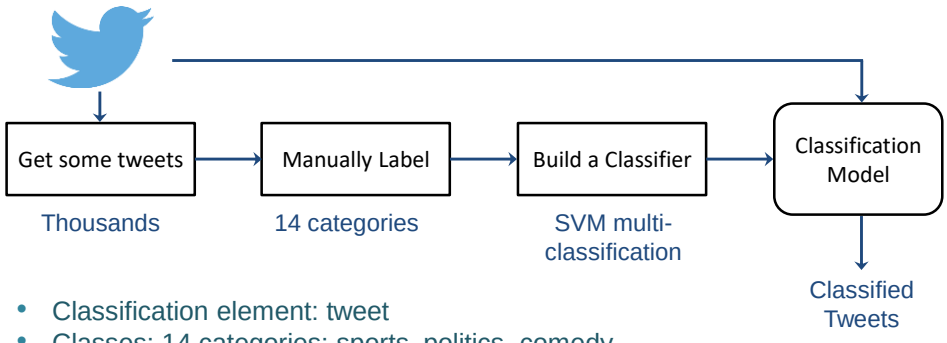
- Implement your first text classifier easy steps
- This is practical lecture
  No equations this time ☺

My first text classifier: Ingredients

- Text elements to be classified
  - Document, paragraph, sentence
- Set of predefined classes (classification task)
  - At least two (binary)
  - Topical, spam, relevance, sentiment, …
- Training set
  - Enough samples of text elements for each class
- Test set
  - Some samples of each class that not used in training
- Features set
  - A set of features extracted from the text to train the classifier
- Classifier
  - The ML module that learns a classification model

My first text classifier: Application

- Classifying tweets into general-purpose categories



- Classification element: tweet
- Classes: 14 categories: sports, politics, comedy, …
- Training/test set: 3129 tweets → 80/20% for train/test
- Features: BOW
- Classifier: SVM multiclass classifier

My first text classifier: Steps

1. Prepare training data
   required: piece of text + label to class
2. Extract features
   1. Pre-process text: lowercase, tokenise, remove useless strings
   2. Create a list of all unique terms in the training data. Give each term a unique ID
   3. Convert the text into features, by replacing each term with its corresponding feature ID. Add value to the feature (simplest value "1" if exists)
   4. For SVM_{light}, zero value features could be neglected. Features need to be listed in order
3. Prepare test file
   1. Convert test file text into features using the same mapping from the training data. For terms that are not in the features list, it could be neglected, or assigned to an ID the represents OOV.
4. Run the learning process on the training data features files to create a model
5. Run the classification on the features file of the test data and save predictions into a file
6. Evaluate performance using accuracy and Macro-F1

Examples

- Tweet + Label

  Kobe passes Wilt for 4th on all-time scoring list Sports

- Learnt features (BOW) from training data

Feature ID	Corresponding work
2944	kobe
2945	rapping
..	
4525	4th
4526	trevi
..	
8330	passes
8331	ducks
..	
9929	17
9930	wilt
...	

- Format after converting text to features

  13 169:1 2944:1 3268:1 4525:1 8063:1 8330:1
  8557:1 8759:1 9116:1 9930:1

  Class ID Feature_ID:Feature_value

- SVM prediction output

  7 -0.546 -0.680 -0.600 -0.411 -0.458 -0.521 4.624
  -0.744 -0.610 -0.687 2.436 -0.612 -0.571 -0.615

  Predicted Class ID

  Score for each of the classes. +ve scores means an acceptable classification (MLMC)

Possible Improvements

- Feature Extraction
  - Apply stemming & stopping
  - Duplicate hashtags words (#car → #car car)
  - Expand tweet text that has link with the page title of that link
  - Add new set of features to the terms appearing in the profile description of the author of the tweet
    - E.g. tweets terms features: ID range: 1 → 12000
      profile terms features: ID range: 12001 → 20000
    - If a term appeared in the tweet and in the profile description, these are two different features with two different IDs
  - Try non-textual features
    - Tweet length, presence of hashtags, links, emojis …

Possible Improvements

- Feature weighting
  - Using tfidf, BM25 as the feature value instead of binary
- Learning method
  - Test other ML learning methods other than SVM
    - Random forest
    - Decision trees
    - Naive Bayesian
  - Test DNNs with word embeddings
- Add more training data
  - Think about a way to create more training data

Resources

- Magdy W., H. Sajjad, T. El-Ganainy and F. Sebastiani. (2015) Bridging Social Media via Distant Supervision.
  Springer SNAM 2015 link, arXiv
- Additional reading:
  Nguyen, D. P., Gravel, R., Trieschnigg, R. B., & Meder, T. How old do you think I am? A study of language and age in Twitter.
  ICWSM 2013