Meetup – Künstliche Intelligenz

6. November 2019

Aspect-based Sentiment Analysis with Machine Learning

Dr. Wiltrud Kessler

Outline

Outline

Motivation

Canon EOS Rebel T3i Digital SLR Camera with EF-S 18-55mm f/3.5-5.6 IS Lens

by Canon

1,973 customer reviews | 912 answered questions

Nikon D3200 24.2 MP CMOS Digital SLR with 18-55mm f/3.5-5.6 AF-S DX NIKKOR Zoom Lens

by Nikon

1,734 customer reviews | 886 answered questions

Motivation

Canon EOS Rebel T3i Digital SLR Camera with EF-S 18-55mm f/3.5-5.6 IS Lens

by Canon

1,973 customer reviews | 912 answered questions

1,973 customer reviews

Nikon D3200 24.2 MP CMOS Digital SLR with 18-55mm f/3.5-5.6 AF-S DX NIKKOR Zoom Lens

by Nikon

1,734 customer reviews | 886 answered questions

1,734 customer reviews

Machine Learning

- We want to automatically analyze reviews.
- We don't want to write lots of rules manually.

Machine Learning

- We want to automatically analyze reviews.
- We don't want to write lots of rules manually.
- We use Machine Learning
  – we let the computer learn the rules from the data.
- The computer can then apply the learned rules to new data.

The task: Sentiment Analysis

Sentiment polarity [Liu 2015]

An opinion is a subjective value statement about an entity. *Sentiment polarity* is the "direction" of the judgment that an opinion expresses.

The task: Sentiment Analysis

Sentiment polarity [Liu 2015]

An opinion is a subjective value statement about an entity. *Sentiment polarity* is the “direction” of the judgment that an opinion expresses.

The Bagels have an outstanding taste with a $\boxed{\text{terrific}}$ texture .

The task: Sentiment Analysis

Sentiment polarity [Liu 2015]

An opinion is a subjective value statement about an entity. *Sentiment polarity* is the "direction" of the judgment that an opinion expresses.

The Bagels have an outstanding taste with a terrific texture .

It is very overpriced and not very tasty .

Machine Learning general concept

Training Data Labels

Training Data

Test Data

Sentences from reviews with sentiment polarity labels

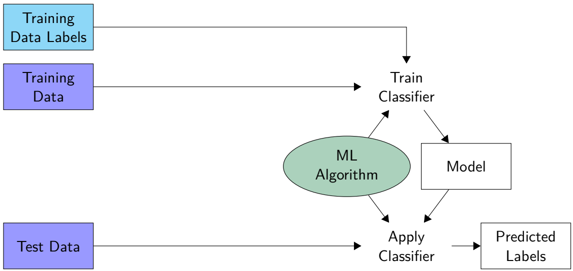
Machine Learning classifier

- We have defined a classification task.
- Classification is a problem from Supervised Machine Learning.
- There are a large number of algorithms for classification.

Machine Learning classifier

- We have defined a classification task.
- Classification is a problem from Supervised Machine Learning.
- There are a large number of algorithms for classification.
- We use Deep Learning for this example, specifically a Recurrent Neural Network
  $\rightarrow$ see next talk for details
- For our purposes, the algorithm is a black box:
  We put in training data and get out a model,
  which we can then apply to new data.

Machine Learning general concept



Sentences from reviews with sentiment polarity labels

Recurrent Neural Network

The input for Machine Learning

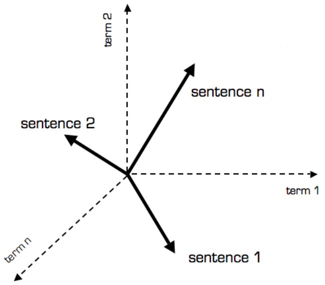
- Problem: Machine Learning classifiers need numbers as input.
- How do we get from text to numbers (vectors)?

The input for Machine Learning

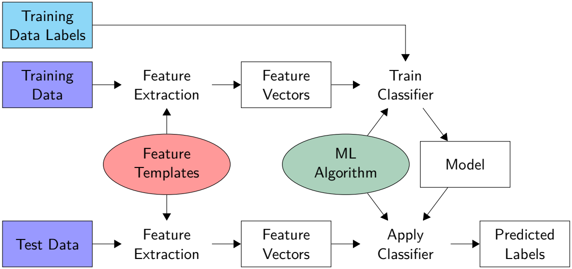
- Problem: Machine Learning classifiers need numbers as input.
- How do we get from text to numbers (vectors)?
- Main idea 1: Bag of words
  - The order of words in a sentence is irrelevant.
  - The same representation for:
    John loves Mary.
    Mary loves John.
    Loves John Mary?

The input for Machine Learning

- Problem: Machine Learning classifiers need numbers as input.
- How do we get from text to numbers (vectors)?
- Main idea 1: Bag of words
  - The order of words in a sentence is irrelevant.
  - The same representation for:
    John loves Mary.
    Mary loves John.
    Loves John Mary?
- Main idea 2: Vector Space Model
  - Every word is a dimension.
  - Every text is a vector.
  - An vector entry is 1 if a word occurs in the text, 0 otherwise.



Machine Learning general concept



Sentences from reviews with sentiment polarity labels
Recurrent Neural Network with bag of words/vector space features

Evaluation

- How do we know if our algorithm works?

Evaluation

- How do we know if our algorithm works?
- We let it run on data where we know the labels and compare the predicted labels with the actual labels.
- This data must be different from the training data (otherwise the algorithm could just memorize all training examples to be perfect)!

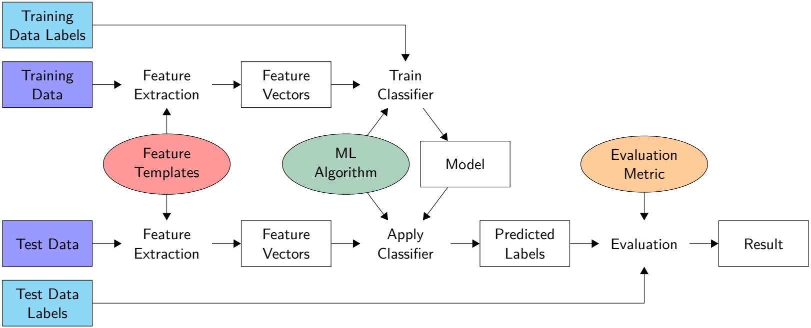
Accuracy for class C

	actually C	actually not C
predicted C	true positives (TP)	false positives (FP)
predicted not C	false negatives (FN)	true negatives (TN)

- Accuracy (A) is the fraction of decisions (C/not C) that are correct:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

Machine Learning general concept



Sentences from reviews with sentiment polarity labels
Recurrent Neural Network with bag of words/vector space features
evaluated by Accuracy

Outline

Training and Test Data

- Restaurant data from SemEval 2014 [Pontiki et al. 2014]
- Training set: 3041 sentences, 3713 category annotations

positive	2179
negative	839
neutral	500
conflict	195

Training and Test Data

- Restaurant data from SemEval 2014 [Pontiki et al. 2014]
- Training set: 3041 sentences, 3713 category annotations

positive	2179
negative	839
neutral	500
conflict	195

Yes, the prices are high , but I felt it was worth it .

Training and Test Data

- Restaurant data from SemEval 2014 [Pontiki et al. 2014]
- Training set: 3041 sentences, 3713 category annotations

positive	2179
negative	839
neutral	500
conflict	195

- Test set: 100 sentences, 114 category annotations

positive	83
negative	21
neutral	9
conflict	1

Yes, the prices are high , but I felt it was worth it .

Let's code!

```
[3] import torch
    import torch.optim as optim
    import torch.nn as nn
    import time
```

```
[4] labeltype = 'term'
    polarities = ['1', '-1']
    use_aspect_label = True
```

```
[5] # Load the data from csv

    ID = data.Field()
    TEXT = data.Field()
    ASPECT = data.Field()
    POLARITY = data.Field()
    LABEL = data.LabelField()

    # field ->   sent.id          text         ex.id          aspect           polarity
    if use_aspect_label:
      fields = [(None, None), ('text', TEXT), (None, None), ('label', LABEL), (None, None)] # Use aspect as label
    else:
      fields = [(None, None), ('text', TEXT), (None, None), (None, None), ('label', LABEL)] # Use polarity as label

    prefix = 'semeval2014_restaurants_' + labeltype + "_" + ".".join(polarities)
    train_data, valid_data, test_data = data.TabularDataset.splits(
                                        path = '/content/drive/My Drive/semeval',
                                        train = prefix + '_train.csv',
                                        validation = prefix + '_val.csv',
                                        test = prefix + '_test.csv',
                                        format = 'csv',
                                        fields = fields,
                                        skip_header = True
    )

    print(f'Number of training examples: {len(train_data)}')
    print(f'Number of validation examples: {len(valid_data)}')
    print(f'Number of testing examples: {len(test_data)}')

    print(vars(train_data.examples[0]))
    print(vars(valid_data.examples[0]))
    print(vars(test_data.examples[0]))
```

```
Number of training examples: 2376
Number of validation examples: 593
Number of testing examples: 86
{'text': ['the', 'first', '2', 'courses', 'were', 'very', 'good', 'but', 'the', 'chocolate', 'sampler', 'was', 'too', 'rich', 'for',
{'text': ['but', 'that', 'wasn', 't', 'the', 'icing', 'on', 'the', 'cake', 'a', 'tiramisu', 'that', 'resembled', 'nothing', 'i', 'ha
{'text': ['all', 'the', 'appetizers', 'and', 'salads', 'were', 'fabulous', 'the', 'steak', 'was', 'mouth', 'watering', 'and', 'the',
```

Outline

The problem with star ratings

THE PROBLEM WITH AVERAGING STAR RATINGS

Aspect-based Sentiment Analysis

Sentiment target and aspects [Liu 2015]

Sentiment does not only have a polarity, but it is also expressed with respect to some *target*, a particular entity or some *aspect* of it.

Aspect-based Sentiment Analysis

Sentiment target and aspects [Liu 2015]

Sentiment does not only have a polarity, but it is also expressed with respect to some *target*, a particular entity or some *aspect* of it.

Overall , excellent restaurant !

Aspect-based Sentiment Analysis

Sentiment target and aspects [Liu 2015]

Sentiment does not only have a polarity, but it is also expressed with respect to some *target*, a particular entity or some *aspect* of it.

Overall , excellent restaurant !

All of the pizzas are terrific and the price is even better !

Aspect-based Sentiment Analysis

Sentiment target and aspects [Liu 2015]

Sentiment does not only have a polarity, but it is also expressed with respect to some *target*, a particular entity or some *aspect* of it.

Overall , excellent restaurant !

All of the pizzas are terrific and the price is even better !

Also very expensive .

Outline

Training and Test Data

- Restaurant data from SemEval 2014 [Pontiki et al. 2014]
- Training set: 3041 sentences, 3713 category annotations

food	1232
service	597
ambience	431
price	321
anecdotes/misc	1132

- Test set: 100 sentences, 114 category annotations

food	41
service	8
ambience	7
price	12
anecdotes/misc	46

Let's code!

```
[3]  import torch
     import torch.optim as optim
     import torch.nn as nn
     import time
```

```
[4]  labeltype = 'term'
     polarities = ['1', '-1']
     use_aspect_label = True
```

```
[5]  # Load the data from csv

     ID = data.Field()
     TEXT = data.Field()
     ASPECT = data.Field()
     POLARITY = data.Field()
     LABEL = data.LabelField()

     # field ->   sent.id          text        ex.id         aspect           polarity
     if use_aspect_label:
       fields = [(None, None), ('text', TEXT), (None, None), ('label', LABEL), (None, None)] # Use aspect as label
     else:
       fields = [(None, None), ('text', TEXT), (None, None), (None, None), ('label', LABEL)] # Use polarity as label

     prefix = 'semeval2014_restaurants_' + labeltype + "_" + ".".join(polarities)
     train_data, valid_data, test_data = data.TabularDataset.splits(
                                         path = '/content/drive/My Drive/semeval',
                                         train = prefix + '_train.csv',
                                         validation = prefix + '_val.csv',
                                         test = prefix + '_test.csv',
                                         format = 'csv',
                                         fields = fields,
                                         skip_header = True
     )

     print(f'Number of training examples: {len(train_data)}')
     print(f'Number of validation examples: {len(valid_data)}')
     print(f'Number of testing examples: {len(test_data)}')

     print(vars(train_data.examples[0]))
     print(vars(valid_data.examples[0]))
     print(vars(test_data.examples[0]))
```

```
Number of training examples: 2376
Number of validation examples: 593
Number of testing examples: 86
{'text': ['the', 'first', '2', 'courses', 'were', 'very', 'good', 'but', 'the', 'chocolate', 'sampler', 'was', 'too', 'rich', 'for',
{'text': ['but', 'that', 'wasn', 't', 'the', 'icing', 'on', 'the', 'cake', 'a', 'tiramisu', 'that', 'resembled', 'nothing', 'i', 'hu
{'text': ['all', 'the', 'appetizers', 'and', 'salads', 'were', 'fabulous', 'the', 'steak', 'was', 'mouth', 'watering', 'and', 'the',
```

Outline

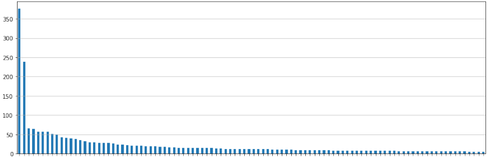
From aspect categories to aspect terms

- Restaurant data from SemEval 2014 [Pontiki et al. 2014]
- Training set: 3041 sentences, 3693 aspect term annotations, 721 different aspect terms
- Test set: 100 sentences, 96 aspect term annotations

From aspect categories to aspect terms

- Restaurant data from SemEval 2014 [Pontiki et al. 2014]
- Training set: 3041 sentences, 3693 aspect term annotations, 721 different aspect terms
- Test set: 100 sentences, 96 aspect term annotations
- Most frequent terms in the training set:

food	376	menu	57
service	238	dinner	56
prices	65	pizza	51
place	64	atmosphere	49
staff	57	price	42



Aspect term variations

- Word forms:
  appetizers (14), *appetizer* (12)

Aspect term variations

- Word forms:
  appetizers (14), *appetizer* (12)
- Orthographic variations:
  ambience (27), *ambiance* (21)

Aspect term variations

- Word forms:
  appetizers (14), *appetizer* (12)
- Orthographic variations:
  ambience (27), *ambiance* (21)
- Misspellings:
  atmosphere (49), *atmoshere* (2), *atomosphere* (2), *atmoshpere* (1), *atmorphere* (1), *at moshphere* (1)

Aspect term variations

- Word forms:
  appetizers (14), *appetizer* (12)
- Orthographic variations:
  ambience (27), *ambiance* (21)
- Misspellings:
  atmosphere (49), *atmoshere* (2), *atomosphere* (2), *atmoshpere* (1), *atmorphere* (1), *at moshphere* (1)
- Synonyms:
  bill (15), *check* (6)

Aspect term variations

- Word forms:
  appetizers (14), *appetizer* (12)
- Orthographic variations:
  ambience (27), *ambiance* (21)
- Misspellings:
  atmosphere (49), *atmoshere* (2), *atomosphere* (2), *atmoshpere* (1), *atmorphere* (1), *at moshphere* (1)
- Synonyms:
  bill (15), *check* (6)
- Abbreviations:
  ac (1), *air conditioning* (1)

Aspect term detail

- Aspects can be very detailed:
  - *margarite pizza with cold prosciutto and baby arugula on top*
  - *wild mushroom third generation fornini pizza*
  - *godmother pizza a sort of traditional flat pizza with an olive oil brushed crust and less tomato sauce than usual*

Aspect term detail

- Aspects can be very detailed:
  - *margarite pizza with cold prosciutto and baby arugula on top*
  - *wild mushroom third generation fornini pizza*
  - *godmother pizza a sort of traditional flat pizza with an olive oil brushed crust and less tomato sauce than usual*
- Aspects form a fine-grained hierarchy:
  food (376)
  appetizers (4)
  cold appetizer dishes (1), *asian appetizers* (1)
  caprese salad appetizer (1), *guacamole shrimp appetizer* (1)

Outline

Bibliography I

[Pontiki et al. 2014] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, S. Manandhar (2014). SemEval-2014 Task 4: Aspect Based Sentiment Analysis. Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014). 27-35. http://alt.qcri.org/semeval2014/task4/

[Liu 2015] Bing Liu (2015). Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Cambridge University Press.

[Zhang et al. 2018] L. Zhang, S. Wang, B. Liu (2018). Deep Learning for Sentiment Analysis: A Survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2018; 8:e1253.

[Trevett 2018] Ben Trevett (2018). PyTorch Sentiment Analysis Tutorial. https://github.com/bentrevett/pytorch-sentiment-analysis

SOFTWARE & SOLUTION DEVELOPER
INNOVATION PARTNER
COFFEE LOVER

Geschäftsführer:
Peter F. Fellinger, Markus Hartinger

Erika-Mann-Straße 63 | 80636 München | +49.89.45 23 47 - 0
Meitnerstraße 8 | 70563 Stuttgart | +49.711.21 95 28 - 0
Klostergasse 3 | 04109 Leipzig | +49.341.22 178 - 0

office@jambit.com
www.jambit.com