NE System Overview
**Learning Algorithm**

**Train Set Feature Generation**

- University, 0, 1, null, Uni, 1, ORG

**Learning Algorithm**

**Trained Machine**

**Test Set Feature Generation**

- USC, 0, 1, 1, 0, null

---

**Given**

<table>
<thead>
<tr>
<th>example</th>
<th>class</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="example1.png" alt="Person" /></td>
<td>PERSON</td>
</tr>
<tr>
<td><img src="example2.png" alt="Organization" /></td>
<td>ORGANIZATION</td>
</tr>
<tr>
<td><img src="example3.png" alt="Person" /></td>
<td>PERSON</td>
</tr>
<tr>
<td><img src="example4.png" alt="Location" /></td>
<td>LOCATION</td>
</tr>
<tr>
<td><img src="example5.png" alt="Organization" /></td>
<td>ORGANIZATION</td>
</tr>
<tr>
<td><img src="example6.png" alt="Location" /></td>
<td>LOCATION</td>
</tr>
<tr>
<td><img src="example7.png" alt="Other" /></td>
<td>OTHER</td>
</tr>
</tbody>
</table>
Learning Algorithm

Train Data

Test Data

Train Set Feature Generation

University,0,1,1,null,Uni,1,ORG

Learning Algorithm

Trained Machine

Test Set Feature Generation

USC,0,1,1,0,null

answer

<table>
<thead>
<tr>
<th>example</th>
<th>Cap.</th>
<th>inDicPer</th>
<th>inDicOrg</th>
<th>inDicLoc</th>
<th>NP</th>
<th>class</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Example 1" /></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>PERSON</td>
</tr>
<tr>
<td><img src="image2.png" alt="Example 2" /></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>ORGANIZATION</td>
</tr>
<tr>
<td><img src="image3.png" alt="Example 3" /></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>PERSON</td>
</tr>
<tr>
<td><img src="image4.png" alt="Example 4" /></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>LOCATION</td>
</tr>
<tr>
<td><img src="image5.png" alt="Example 5" /></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>ORGANIZATION</td>
</tr>
<tr>
<td><img src="image6.png" alt="Example 6" /></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>LOCATION</td>
</tr>
<tr>
<td><img src="image7.png" alt="Example 7" /></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>OTHER</td>
</tr>
</tbody>
</table>

nxm matrix, where n is number of examples, m is number of features+class label
Choose a machine learning classifier from Weka
Given

<table>
<thead>
<tr>
<th>example</th>
<th>class</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>IBM</td>
<td>?</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

Note that the class is unknown for the examples of the test data
<table>
<thead>
<tr>
<th>example</th>
<th>Cap.</th>
<th>inDicPer</th>
<th>inDicOrg</th>
<th>inDicLoc</th>
<th>NP</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="example" /></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td><img src="image2.png" alt="example" /></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td><img src="image3.png" alt="example" /></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td><img src="image4.png" alt="example" /></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><img src="image5.png" alt="example" /></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><img src="image6.png" alt="example" /></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### Table:

<table>
<thead>
<tr>
<th>example</th>
<th>Cap.</th>
<th>inDicPer</th>
<th>inDicOrg</th>
<th>inDicLoc</th>
<th>NP</th>
<th>Predicted Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="example.png" alt="example" /></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>LOCATION</td>
</tr>
<tr>
<td><img src="example.png" alt="example" /></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>LOCATION</td>
</tr>
<tr>
<td><img src="example.png" alt="example" /></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>PERSON</td>
</tr>
<tr>
<td><img src="example.png" alt="example" /></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>ORGANIZATION</td>
</tr>
<tr>
<td><img src="example.png" alt="example" /></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>OTHER</td>
</tr>
<tr>
<td><img src="example.png" alt="example" /></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>OTHER</td>
</tr>
</tbody>
</table>

### Precision

$$\text{Precision} = \frac{\# \text{ correct identified NEs}}{\# \text{ identified NEs}}$$

### Recall

$$\text{Recall} = \frac{\# \text{ correct identified NEs}}{\# \text{ gold standard data}}$$
NE Feature Generation
Features (1)

- **Contextual**
  - current word $W_0$
  - words around $W_0$ in [-3,...,+3] window

- **Part-of-speech tag** (when available)

- **Orthographic**
  
  - `initial-caps`
  - `roman-number`
  - `acronym`
  - `single-char`
  - `all-caps`
  - `contains-dots`
  - `lonely-initial`
  - `functional-word*`
  - `all-digits`
  - `contains-hyphen`
  - `punctuation-mark`
  - `URL`

- **Word-Type Patterns**
  
  - `functional`
  - `lowercased`
  - `punctuation mark`
  - `quote`
  - `capitalized`
  - `other`

- **Left Predictions**
  
  - the tag predicted in the current classification for $W_{-3}$, $W_{-2}$, $W_{-1}$

*`functional-word is preposition, conjunction, article*
Features (2)

• **Bag-of-Words**
  • words in [-5,...,+5] window

• **Trigger words**
  • for person *(Mr., Miss., Dr., PhD.)*
  • for location *(city, street)*
  • for organization *(Ltd., Co.)*

• **Gazetteers**
  • names of cities, countries, villages, streets
  • names of organizations
  • person first name
  • person surname

* put each type of trigger words and gazetteers in separate files, because you can treat them as separate features
Features (3)

• Length in words of the entity being classified

• Pattern of the entity with regard to the type of constituent words

• **For each classs**
  • whole NE is in gazetteer
  • any component of the NE appears in gazetteer

• **Suffixes** (length 1 to 4)

• Previous word is an article

• Previous word is a noun

• More idea on features:
Collecting External Resources
Gazetteer Collection Method 1

• Yago contains over 2 million entities (like persons, organizations, cities among others)

• Download Yago from:


• Extract from the relevant relations all named entities

Ex.

– X born in Y, where X is a person and Y is a location
– X works for Y, where X is a person and Y is a person or an organization
Madonna (entertainer)

From Wikipedia, the free encyclopedia

Madonna (born Madonna Louise Ciccone; August 16, 1958) is an American recording artist, actress and entrepreneur. Born in Bay City, Michigan, and raised in Rochester Hills, Michigan, she moved to New York City in 1977, for a career in modern dance. After performing as a member of the pop musical groups Breakfast Club and Emmy, she released her self-titled debut album, Madonna, in 1983 on Sire Records.

A series of hit singles from her next studio albums, Like a Virgin (1984) and True Blue (1986), gained her global recognition. They established her as a pop icon, for pushing the boundaries of lyrical content in mainstream popular music and imagery in her music videos, which became a fixture on MTV. Her recognition was augmented by the film Desperately Seeking Susan (1985) which widely became seen as a Madonna vehicle, despite her not playing the lead. Expanding on the use of religious imagery with Like a Prayer (1989), Madonna received positive critical reception for her diverse musical productions, while at the same time was criticised by religious conservatives and the Vatican. In 1992, Madonna founded the Maverick corporation, a joint venture between herself and Time Warner. The same year, she expanded the use of sexually explicit material in her work, beginning with the release of the studio album Erotica, followed by the publishing of the coffee table book Sex, and starring in the erotic thriller Body of Evidence, all of which received negative responses from conservatives and liberals alike.

In 1998, Madonna played the starring role in the film Evita, for which she won a Golden Globe Award for Best Actress in Motion Picture Musical or Comedy. Madonna’s seventh studio album, Ray of Light (1998), became one of her most critically acclaimed, recognized for its lyrical depth. During the 2000s, Madonna released four studio albums – namely Music (2000), American Life (2003), Confessions on a Dance Floor (2005) and Hard Candy (2008) – all of which debuted at number one on the Billboard 200. Departing from Warner Bros. Records, Madonna signed an unprecedented $120 million dollar contract with Live Nation in 2008.

According to the International Federation of the Phonographic Industry, Madonna has sold more than 200 million albums worldwide. She is ranked by the Recording Industry Association of America as the best-selling female rock artist of the 20th century, and the second top-selling female artist in the United States, behind Barbra Streisand, with 64 million certified albums. Guinness World Records listed her as the world's most successful female recording artist of all time. In 2008, Billboard magazine ranked Madonna at number two, behind only The Beatles, on the "Billboard Hot 100 All-Time Top Artists", making her the most successful solo artist in the history of the chart. She was also inducted into the Rock and Roll Hall of Fame in the same year. Considered to be one of the most influential women in contemporary music, Madonna has been known for continually reinventing both her music and image, and for retaining a standard of autonomy within the recording industry. She is recognized as an influence among numerous music artists.
Gazetteer Collection Method 2

• Step 1: Check if identified NE exists in Wikipedia

• Step 2: Extract the first 2-3 sentences

• Step 3: Pull the nouns matching the expression
  
  X is Y, Z  
  X is Y and Z

• Step 4: Extract the information from the infobox

• Step 5: Verify in WordNet whether the found concepts are hyponyms of person, location, organization

(Madonna is an artist, actress)
Gazetteer Collection Method 3

- contains structured information from Wikipedia

Class | Instance
--- | ---
Place | 462,000
Person | 364,000
Organization | 148,000
Resource (overall) | 1,667,000

http://wiki.dbpedia.org/Datasets#h18-11
Other Gazetteer Sources

- The 2000 U.S. Census data
  
  http://www.rdfabout.com/demo/census/

- Freebase
  
  http://www.freebase.com/schema/people

- Linked Data Sets
  
  http://esw.w3.org/DataSetRDFDumps

...
Patterns
Pattern Extraction

• Collect statistics for patterns containing NEs
  Ex.
  – Jenny_PER works_O for_O IBM_ORG .O
  – Sam_PER works_O for_O Microsoft_ORG .O
  – Paul_PER Adams_PER worked_O for_O George_PER .O
  – Jenny_PER bought_O an_O organge_O .O
  – Yahoo!_ORG bought_O Overtrue_ORG .O

• Extract verbs to the left and to the right of the NE
  Ex.
  – London_LOC is_O located_O in_O
  – John_PER drinks_O juice_O
Classifier Combination
Majority Voting

- Let $C^1 \ldots C^N$ be the set of classifiers that are induced by training $N$ different learning algorithms $L^1 \ldots L^N$ on a data set $D$ consisting of feature vectors.
- Given a new instance, query classifiers $C^1 \ldots C^N$ and assign to the instance the class with the highest count

<table>
<thead>
<tr>
<th>$C^1$</th>
<th>$C^2$</th>
<th>$C^3$</th>
<th>VOTING</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOC</td>
<td>LOC</td>
<td>PER</td>
<td>LOC</td>
</tr>
<tr>
<td>PER</td>
<td>ORG</td>
<td>PER</td>
<td>PER</td>
</tr>
<tr>
<td>ORG</td>
<td>LOC</td>
<td>PER</td>
<td>ORG</td>
</tr>
</tbody>
</table>

Expected question by Cris: **can we do weighted voting?**

- Look at weighted voting and voting with probability distribution (Diettrich, 1997)
10-fold Cross Validation

- Data is split into 10 approximately equal partitions
- Each partition is used in turn for testing while the remainder is used for training
  - 9/10 of data is used for training
  - 1/10 of the data is used for testing
- Repeat the whole procedure 10 times
- Overall error rate is equal to the average of the error rates on each partition
- Finally generate the final classifier by learning from all of the data.
Stacking

- Learn a meta (level-1) classifier using the output of base-level (level-0) classifiers estimated via cross-validation.

![Diagram of Stacking process]

- Base-level data set $D$
  - $D^K$ and $D \setminus D^K$
  - $C^1(k) \ldots C^N(k)$
  - Feature vectors
  - Meta-level data set $MD^K$

- Meta-level data set $MD$
  - $MD^K$

- Level-1 to $N$ data sets $L^1 \ldots L^N$
  - New instance $X$

- Level $M$ data set $L^M$
  - Level $M$ classifier $C^M$
  - Class value $Y$
Boosting and Bagging

• Let $C^1 \ldots C^N$ be the set of classifiers that are generated by applying a single learning algorithm to $N$ different versions of a given data set, rather than training $N$ different algorithms.

• Typically examples that are misclassified gain weight and examples that are classified correctly lose weight.

• Relevant literature
  – Boosting (Freud and Schapire, 1996)
  – Bagging (Breiman, 1996)
Amount of Training Data
Effect of Training Data Size

- Study the effect of the number of examples used during training and the performance of the classifier $C^j$
Semi-Supervised Learning
Semi-supervised Learning

• Learn from a small amount of labeled data and a large amount of unlabeled data

• Methods:
  – Self-training [E.g. Banko & Brill, 2001, Nigham]
  – Active learning [E.g. Cohn et al. 1994; Lewis & Catlett 1994; Schohn & Cohn 2000; Shen 2004]

• Tasks which could be resolved: NE recognition, POS tagging, Parsing, ...
Co-training / Self-training

- A set $L$ of labeled training examples
- A set $U$ of unlabeled examples
- Classifiers $C_i$

1. Create a pool of examples $U'$
   - choose $P$ random examples from $U$
2. Loop for $I$ iterations
   - Train $C_i$ on $L$ and label $U'$
   - Select $G$ most confident examples and add to $L$
     - maintain distribution in $L$
   - Refill $U'$ with examples from $U$
     - keep $U'$ at constant size $P$
Co-training/Self-training
WEKA
Waikato Environment for Knowledge Analysis
Weka: Data Mining Software

• Collection of machine learning algorithms
  – open-source package written in Java

• Used for research, education and application

• Main features:
  – data pre-processing tools
  – learning algorithms
  – evaluation methods
  – graphical inference
  – environment for comparing learning algorithms
Weka: Data Mining Software

• Classification algorithms:
  – decision trees, linear classifiers, SVM, Naive-bayes, kNN

• Prediction algorithms:
  – regression (linear/SVM), perceptron

• Meta-algorithms:
  – bagging, boosting (AdaBoost)

among others
Getting Started

• Install Weka software (on Linux):
  – Download link:
    • http://prdownloads.sourceforge.net/weka/weka-3-6-2.zip
    • Unzip the software
  – Requirement: Java 1.5 (or higher)
  – Invoke Weka command:
    • java -cp weka.jar <weka-command>
Weka GUI Chooser

```
java -Xmx1000M -jar weka.jar
```
Data file format (.arff)

@relation english_named_entity

@attribute position numeric
@attribute pos_tag {NN, NP, VB, DT}
@attribute word_length numeric
@attribute in_gazetteer {no, yes}
@attribute class {PER, LOC, ORG, MISC}

@data
3,DT,3,no,ORG
4,NP,10,yes,ORG
15,NP,6,yes,PER
7,NN,12,?,MISC
...

Other attribute types:
• String
• Date

Missing value
The Preprocessing Tab

- Classification
- Statistical attribute selection
- Filter selection
- Manual attribute selection
- List of attributes (last: class variable)
- Frequency and categories for the selected attribute
- Statistics about the values of the selected attribute

Slide adapted from Marti Hearst
Choice of classifier

The attribute whose value is to be predicted from the values of the remaining ones. Default is the last attribute.

Cross-validation: split the data into e.g. 10 folds and 10 times train on 9 folds and test on the remaining one.
Choosing a classifier

Slide adapted from Marti Hearst
Slide adapted from Marti Hearst
all other numbers can be obtained from it

different/easy class

accuracy

TP Rate  FP Rate  Precision  Recall  F-Measure  Class
0.75     0.3      0.556     0.75     0.638      misc.forsale
0.7      0.025    0.933     0.7      0.8        rec.sport.hockey
0.6      0.15     0.667     0.6      0.632      comp.graphics

Confusion Matrix:

a  b  c  <-- classified as
15 1  4  a = misc.forsale
14 2  1  b = rec.sport.hockey
 8 0 12  c = comp.graphics
Running on Test Set

Correctly Classified Instances

<table>
<thead>
<tr>
<th>Class</th>
<th>Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>396</td>
</tr>
<tr>
<td>b</td>
<td>128</td>
</tr>
</tbody>
</table>

Incorrectly Classified Instances

<table>
<thead>
<tr>
<th>Class</th>
<th>Incorrect</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>2</td>
</tr>
<tr>
<td>b</td>
<td>271</td>
</tr>
</tbody>
</table>

Total Number of Instances

218

Detailed Accuracy by Class

<table>
<thead>
<tr>
<th>Class</th>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0.995</td>
<td>0.321</td>
<td>0.756</td>
<td>0.993</td>
</tr>
<tr>
<td>b</td>
<td>0.679</td>
<td>0.005</td>
<td>0.993</td>
<td>0.995</td>
</tr>
</tbody>
</table>

Confusion Matrix

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>396</td>
<td>2</td>
</tr>
<tr>
<td>b</td>
<td>128</td>
<td>271</td>
</tr>
</tbody>
</table>
WEKA
Command Line
Weka specifications

• Train classifier on training data and output model
  • java -cp weka.jar <classifier-function> –t <train-file> -d <trained-model>

• Run trained classifier model on test data
  • java -cp weka.jar <classifier-function> –T <test-file> -l <trained-model>

• Specifying parameters:
  -t : training file (.arff)
  -T : test file (.arff)
  -d : output filename (trained classifier model)
  -l : input model (for testing)
  -K : number of nearest neighbors for kNN algorithm
  -h : help (check out other parameter options, etc.)
Example: kNN in Weka

• Train a classifier using 2NN algorithm
  • java -cp weka.jar
    weka.classifiers.lazy.IBk
    -t data/weather.arff
    -K 2
    -d model.2nn

Classifier-function in weka
Training file
Algorithm parameter
Output model name

• Run the trained classifier on test data
  • java -cp weka.jar
    weka.classifiers.lazy.IBk
    -T data/weather.arff
    -l model.2nn

Classifier-function in weka
Test file
Input model name
## Sample Weka output

### Error on test data

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Classified Instances</td>
<td>13</td>
<td>92.8571 %</td>
</tr>
<tr>
<td>Incorrectly Classified Instances</td>
<td>1</td>
<td>7.1429 %</td>
</tr>
<tr>
<td>Kappa statistic</td>
<td>0.8372</td>
<td></td>
</tr>
<tr>
<td>Mean absolute error</td>
<td>0.1333</td>
<td></td>
</tr>
<tr>
<td>Root mean squared error</td>
<td>0.2333</td>
<td></td>
</tr>
<tr>
<td>Total Number of Instances</td>
<td>14</td>
<td></td>
</tr>
</tbody>
</table>
More detailed output

- Classification labels for each instance (use "–p 1" option)
  - `java -cp weka.jar weka.classifiers.lazy.Ibk -T data/weather.arff -l model.2nn -p 1`

```plaintext
=== Predictions on test data ===

<table>
<thead>
<tr>
<th>inst#</th>
<th>actual</th>
<th>predicted</th>
<th>error</th>
<th>prediction (outlook)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2:no</td>
<td>2:no</td>
<td>0.967</td>
<td>sunny</td>
</tr>
<tr>
<td>2</td>
<td>2:no</td>
<td>1:yes</td>
<td>0.5</td>
<td>sunny</td>
</tr>
<tr>
<td>3</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>overcast</td>
</tr>
<tr>
<td>4</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>rainy</td>
</tr>
<tr>
<td>5</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>rainy</td>
</tr>
<tr>
<td>6</td>
<td>2:no</td>
<td>2:no</td>
<td>0.967</td>
<td>rainy</td>
</tr>
<tr>
<td>7</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>overcast</td>
</tr>
<tr>
<td>8</td>
<td>2:no</td>
<td>2:no</td>
<td>0.967</td>
<td>sunny</td>
</tr>
<tr>
<td>9</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.5</td>
<td>sunny</td>
</tr>
<tr>
<td>10</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>rainy</td>
</tr>
<tr>
<td>11</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.5</td>
<td>sunny</td>
</tr>
<tr>
<td>12</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>overcast</td>
</tr>
<tr>
<td>13</td>
<td>1:yes</td>
<td>1:yes</td>
<td>0.967</td>
<td>overcast</td>
</tr>
<tr>
<td>14</td>
<td>2:no</td>
<td>2:no</td>
<td>0.967</td>
<td>rainy</td>
</tr>
</tbody>
</table>
```
Weka classification functions

- kNN: `weka.classifiers.lazy.Ibk`
- Decision trees: `weka.classifiers.trees.J48`
- Naïve Bayes: `weka.classifiers.bayes.NaiveBayes`
- AdaBoost: `weka.classifiers.meta.AdaBoostM1`
Additional Information

• General documentation:
  http://www.cs.waikato.ac.nz/ml/weka/
  http://prdownloads.sourceforge.net/weka/weka.ppt

• Command line doc:
  http://weka.wikispaces.com/Primer
Remember to send an e-mail to

kozareva@isi.edu

with subject **CS544 homework**

to obtain the train and development data for Assignment 1