Predictive Analysis of Text: Concepts, Instances, and Classifiers

Heejun Kim

May 29, 2018
Predictive Analysis of Text

• **Objective**: developing computer programs that automatically **predict** a particular **concept** within a span of text
## Predictive Analysis: Procedure

### Model

**Test Data**

<table>
<thead>
<tr>
<th>color</th>
<th>size</th>
<th>sides</th>
<th>equal sides</th>
<th>...</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>no</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>green</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>inf</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>4</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
</tbody>
</table>

### Representation

<table>
<thead>
<tr>
<th>color</th>
<th>size</th>
<th>sides</th>
<th>equal sides</th>
<th>...</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>no</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>green</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>inf</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>4</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
</tbody>
</table>
Predictive Analysis: basic ingredients

• **Training data**: a set of examples of the labeled concept we want to automatically recognize

• **Representation**: a set of features that we believe are useful in recognizing the desired concept

• **Learning algorithm**: a computer program that uses the training data to learn a predictive model of the concept
Predictive Analysis: basic ingredients

- **Model**: a function that describes a predictive relationship between feature values and the presence/absence of the concept
- **Test data**: a set of previously unseen examples used to estimate the model’s effectiveness
- **Performance metrics**: a set of statistics used to measure the predictive effectiveness of the model
Predictive Analysis: training and testing

- labeled examples

- machine learning algorithm

- model

- testing

- new, unlabeled examples

- predictions
## Predictive Analysis:
concept, instances, and features

<table>
<thead>
<tr>
<th>instances</th>
<th>color</th>
<th>size</th>
<th># slides</th>
<th>equal sides</th>
<th>...</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>no</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>green</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>inf</td>
<td>yes</td>
<td></td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>4</td>
<td>yes</td>
<td></td>
<td>...</td>
<td>no</td>
</tr>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td></td>
<td>yes</td>
</tr>
</tbody>
</table>
Predictive Analysis: 
Type of features

• Nominal: values that are distinct symbols (e.g., male and female). No ordering or distance.

• Numeric
  – Ordinal: ranked order of the categories (e.g., hot, mild, and cool). No distance.
  – Interval: ordered and measured in fixed and equal units (e.g., temperature and school year). 0 is arbitrary.
  – Ratio: measurement method inherently defines a zero point (e.g., distance). Ordered and measured in fixed and equal units.
Predictive Analysis: training and testing

<table>
<thead>
<tr>
<th>color</th>
<th>size</th>
<th># slides</th>
<th>Equal sides</th>
<th>...</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>no</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>green</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>inf</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>4</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
</tbody>
</table>

labeled examples

training

model

machine learning algorithm

testing

model

new, unlabeled examples

<table>
<thead>
<tr>
<th>color</th>
<th>size</th>
<th># slides</th>
<th>Equal sides</th>
<th>...</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>no</td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>green</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>inf</td>
<td>yes</td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>4</td>
<td>yes</td>
<td></td>
<td>?</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>

predictions

<table>
<thead>
<tr>
<th>color</th>
<th>size</th>
<th># slides</th>
<th>Equal sides</th>
<th>...</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>no</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>green</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>inf</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>blue</td>
<td>small</td>
<td>4</td>
<td>yes</td>
<td></td>
<td>no</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>red</td>
<td>big</td>
<td>3</td>
<td>yes</td>
<td></td>
<td>yes</td>
</tr>
</tbody>
</table>
Predictive Analysis: questions

- Is a particular concept appropriate for predictive analysis?
- What should the unit of analysis be?
- How should I divide the data into training and test sets?
- What is a good feature representation for this task?
- What type of learning algorithm should I use?
- How should I evaluate my model’s performance?
Predictive Analysis: Concepts

• Learning algorithms can recognize some concepts better than others
• What are some properties of concepts that are easier to recognize?
Predictive Analysis: Concepts

- Option 1: can a human recognize the concept?
- Option 2: can two or more humans recognize the concept independently and do they agree?
- Option 2 is better.
- In fact, models are sometimes evaluated as an independent assessor.
- How does the model’s performance compare to the performance of one assessor with respect to another?
  - One assessor produces the “ground truth” and the other produces the “predictions”
Predictive Analysis:
measures agreement: percent agreement

- **Percent agreement**: percentage of instances for which both assessors agree that the concept occurs or does not occur

\[
\frac{(A + D)}{(A + B + C + D)}
\]
Predictive Analysis: measures agreement: percent agreement

- Problem: percent agreement does not account for agreement due to random chance.
- How can we compute the expected agreement due to random chance?
Predictive Analysis: measures agreement: percent agreement

- Percent agreement:
  \[
  \frac{(80 + 10)}{(80 + 5 + 5 + 10)}
  \]

- Agreement due to random chance?
Predictive Analysis: measures agreement: percent agreement

• How can we compute the expected agreement due to random chance?

• **Kappa agreement:** percent agreement after correcting for the expected agreement due to chance (not covered in this course)

• For more details, refer to [Wikipedia article](https://en.wikipedia.org) or [online video](https://www.youtube.com)
Predictive Analysis: questions

- Is a particular concept appropriate for predictive analysis?
- **What should the unit of analysis be?**
- How should I divide the data into training and test sets?
- What is a good feature representation for this task?
- What type of learning algorithm should I use?
- How should I evaluate my model’s performance?
Predictive Analysis: turning data into training and test instances

• For many text-mining applications, turning the data into instances for training and testing is fairly straightforward

• Easy case: instances are self-contained, independent units of analysis

• topic categorization: instances = documents

• opinion mining: instances = product reviews

• bias detection: instances = political blog posts

• emotion detection: instances = support group posts
# Topic Categorization: predicting health-related documents

<table>
<thead>
<tr>
<th>instances</th>
<th>features</th>
<th>concept</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>w_1</td>
<td>w_2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Opinion Mining
predicting positive/negative movie reviews

### features

<table>
<thead>
<tr>
<th>w_1</th>
<th>w_2</th>
<th>w_3</th>
<th>...</th>
<th>w_n</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>...</td>
<td>0</td>
<td>positive</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>...</td>
<td>0</td>
<td>negative</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>...</td>
<td>0</td>
<td>negative</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>...</td>
<td>1</td>
<td>negative</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>0</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>...</td>
<td>1</td>
<td>positive</td>
</tr>
</tbody>
</table>

### concept
Bias Detection
predicting liberal/conservative blog posts

<table>
<thead>
<tr>
<th>features</th>
<th>concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>w_1</td>
<td>label</td>
</tr>
<tr>
<td>w_2</td>
<td></td>
</tr>
<tr>
<td>w_3</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>w_n</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>instances</th>
<th>liberal</th>
<th>conservative</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 1 0 ... 0</td>
<td></td>
<td>liberal</td>
</tr>
<tr>
<td>0 0 0 ... 0</td>
<td></td>
<td>conservative</td>
</tr>
<tr>
<td>0 0 0 ... 0</td>
<td></td>
<td>conservative</td>
</tr>
<tr>
<td>0 1 0 ... 1</td>
<td></td>
<td>conservative</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>1 0 0 ... 1</td>
<td></td>
<td>liberal</td>
</tr>
</tbody>
</table>
Predictive Analysis: questions

• Is a particular concept appropriate for predictive analysis?
• What should the unit of analysis be?
• How should I divide the data into training and test sets?
• What is a good feature representation for this task?
• What type of learning algorithm should I use?
• How should I evaluate my model’s performance?
Predictive Analysis: training and test data

• We want our model to “learn” to recognize a concept

• So, what does it mean to learn?
Predictive Analysis: training and test data

• The machine learning definition of learning:

A machine *learns* with respect to a particular task $T$, performance metric $P$, and experience $E$, if the system improves its *performance* $P$ at task $T$ following experience $E$.

-- Tom Mitchell
Predictive Analysis: can we use the same data for testing?

Training Data → training → machine learning algorithm → Spam Detection Model

Test Data

New Data
Predictive Analysis: training and test data

• We want our model to improve its generalization performance!
• That is, its performance on previously unseen data!
• **Generalize**: to derive or induce a general conception or principle from particulars. -- Merriam-Webster
• In order to test generalization performance, the training and test data cannot be the same.
• Why?
Training data + Representation: what could possibly go wrong?
Predictive Analysis: training and test data

• While we don’t want to test on training data, we want to have training and test set that are derived from the same “probability distribution”.

• What does that mean?
Predictive Analysis: training and test data

Data

Training Data

Test Data

: positive instances

: negative instances
Predictive Analysis: training and test data

• Is this a good partitioning? Why or why not?
Predictive Analysis: training and test data

Data

Training Data

Test Data

Random Sample

Random Sample

: positive instances

: negative instances
Predictive Analysis: training and test data

- On average, random sampling should produce comparable data for training and testing.

Data

- Positive instances
- Negative instances

Training Data

Test Data

: positive instances
: negative instances
Statistical Estimation
Predictive Analysis: training and test data

1. Split data into training & testing subsets
2. Train a model on training set
3. Make predictions on the testing set
4. Compare predicted and true labels
Predictive Analysis: training and test data

- If you want to predict stock price by analyzing tweets, how the training and test data should be separated?

![Diagram showing time points for training and test data]

- $t_0, t_1, t_2, t_3, t_4$
If you want to predict stock price by analyzing tweets, how the training and test data should be separated?

Predictive Analysis: training and test data

- Training data
  - $t_0$, $t_1$, $t_2$, $t_3$
- Test data
  - $t_4$
Predictive Analysis: training and test data

- Models usually perform the best when the training and test set have:
  - a similar proportion of positive and negative examples
  - a similar co-occurrence of feature-values and each target class value
Predictive Analysis: training and test data

• Caution: in some situations, partitioning the data randomly might inflate performance in an unrealistic way!
• How the data is split into training and test sets determines what we can claim about generalization performance
• The appropriate split between training and test sets is usually determined on a case-by-case basis
Predictive Analysis: discussion

- **Spam detection:** should the training and test sets contain email messages from the same sender, same recipient, and/or same timeframe?
- **Topic segmentation:** should the training and test sets contain potential boundaries from the same discourse?
- **Opinion mining for movie reviews:** should the training and test sets contain reviews for the same movie?
- **Sentiment analysis:** should the training and test sets contain blog posts from the same discussion thread?
Predictive Analysis: questions

- Is a particular concept appropriate for predictive analysis?
- What should the unit of analysis be?
- How should I divide the data into training and test sets?
- What is a good feature representation for this task?
- **What type of learning algorithm should I use?**
- How should I evaluate my model’s performance?
Predictive Analysis:
three types of classifiers

• Linear classifiers
• Decision tree classifiers
• Instance-based classifiers
Predictive Analysis:
three types of classifiers

• All types of classifiers learn to make predictions based on the input feature values

• However, different types of classifiers combine the input feature values in different ways
Predictive Analysis:
three types of classifiers

\[ y = \begin{cases} 
1 & \text{if } w_0 + \sum_{j=1}^{n} w_j x_j > 0 \\
0 & \text{otherwise} 
\end{cases} \]
Learning Algorithm + Model: what could possibly go wrong?

Relationship between Usefulness and word count
Predictive Analysis
linear classifiers: perceptron algorithm

\[ y = \begin{cases} 
1 & \text{if } w_0 + \sum_{j=1}^{n} w_j x_j > 0 \\
0 & \text{otherwise} 
\end{cases} \]

parameters learned by the model
predicted value (e.g., 1 = positive, 0 = negative)
Predictive Analysis
linear classifiers: perceptron algorithm

\[
y = \begin{cases} 
1 & \text{if } w_0 + \sum_{j=1}^{n} w_j x_j > 0 \\
0 & \text{otherwise}
\end{cases}
\]

test instance

<table>
<thead>
<tr>
<th>f_1</th>
<th>f_2</th>
<th>f_3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

model weights

<table>
<thead>
<tr>
<th>w_0</th>
<th>w_1</th>
<th>w_2</th>
<th>w_3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>-5</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

output = \(2.0 + (0.50 \times -5.0) + (1.0 \times 2.0) + (0.2 \times 1.0)\)

output = 1.7

output prediction = positive
Predictive Analysis
linear classifiers: perceptron algorithm

(two-feature example borrowed from Witten \textit{et al.} textbook)
Predictive Analysis
linear classifiers: logistic regression

\[
\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}
\]

when \( t = \beta_0 + \beta_1 x \)

Figure 1. The standard logistic function \( \sigma(t) \); note that \( \sigma(t) \in (0, 1) \) for all \( t \).

Predictive Analysis: would a linear classifier work?
Predictive Analysis: three types of classifiers

• Linear classifiers
• Decision tree classifiers
• Instance-based classifiers
Predictive Analysis
decision tree classifiers
Predictive Analysis
decision tree classifiers

• Decision Tree
  – Special decision rules organized in form of tree data structure that help to understand the relationship among the attributes and class labels.
  – Attributes become nodes, edges are used to represent the values of these attributes, and predictions are made at each leaf.
Predictive Analysis: decision tree classifiers

- Draw a decision tree that would perform perfectly on this training data!
Predictive Analysis: examples of decision tree classifiers
Predictive Analysis: three types of classifiers

- Linear classifiers
- Decision tree classifiers
- Instance-based classifiers
Predictive Analysis: instance-based classifiers

- predict the class associated with the most similar training examples
Predictive Analysis: instance-based classifiers

- predict the class associated with the most similar training examples
Predictive Analysis: instance-based classifiers

- **Assumption**: instances with similar feature values should have a similar label
- Given a test instance, predict the label associated with its nearest neighbors
- There are many different similarity metrics for computing distance between training/test instances
Predictive Analysis: questions

• Is a particular concept appropriate for predictive analysis?
• What should the unit of analysis be?
• How should I divide the data into training and test sets?
• What is a good feature representation for this task?
• What type of learning algorithm should I use?
• How should I evaluate my model’s performance?
Any Questions?
Next Class