Web Mining

Web Content Mining
- Part 1 -

Simone Paolo Ponzetto
Stefano Faralli

FSS 2017
Web Content Mining

■ Definition

Automatic extraction of useful information (facts, patterns) from Web content (text, images, multimedia).

■ Content Mining Tasks
  ■ Content Clustering
  ■ Content Classification
  ■ Information Extraction
  ■ Sentiment Analysis

Main focus of this lecture / topic block

Will be discussed as subtask of sentiment analysis
The basic task in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature/aspect level.

**Polarity Values**
- Positive, neutral, negative
- Likert scale (1 to 10)

**Application Examples**
- **Document-Level**
  - tweet analysis about politicians
- **Feature/Aspect-Level**
  - analysis of product reviews
Information Extraction

Information extraction (IE) is the task of automatically extracting structured information from unstructured or semi-structured machine-readable documents.

- **Subtasks**
  - **Named Entity Recognition and Disambiguation**
    - “M. Smith likes fishing“
    - Which M. Smith?
  - **Coreference Resolution**
    - “M. Smith likes fishing. But he doesn't like biking.”
    - Does he refer to M. Smith?
  - **Relationship Extraction**
    - PERSON works for ORGANIZATION
    - PERSON located in LOCATION
Roadmap

- Introduction
- Opinion mining – problem definition
- Document level sentiment classification
- Aspect-based opinion mining
Literature


- Further reading and lots of practical pointers to corpora and resources: Opinion mining and sentiment analysis. NOW. http://www.cs.cornell.edu/home/llee/opinion-mining-sentiment-analysis-survey.html

- This first part of this slide deck is heavily based on online materials and presentations from Bing Liu (thanks!).
Introduction – user generated content

Word-of-mouth on the Web

- One can express personal experiences and opinions on almost anything, at review sites, forums, discussion groups, blogs ... (called the user generated content.)

- They contain valuable information

Web/global scale: No longer – one’s circle of friends

Our interest: to mine opinions (sentiments) expressed in the user-generated content

- An intellectually very challenging problem.
- Practically very useful.
Introduction – Applications

- **Businesses and organizations**: product and service benchmarking. Market intelligence.
  - Business spends a huge amount of money to find consumer sentiments and opinions.
    - Consultants, surveys and focus groups, etc.

- **Individuals**: interested in other’s opinions when
  - Purchasing a product or using a service,
  - Finding opinions on political topics,

- **Ads placements**: Placing ads in the user-generated content
  - Place an ad when one praises a product.
  - Place an ad from a competitor if one criticizes a product.

- **Opinion retrieval/search**: providing general search for opinions.
Two types of evaluation

- **Regular Opinions**: sentiment expressions on some entities, e.g., products, events, topics, persons.
  - E.g., “the picture quality of this camera is great”
  - Subjective

- **Comparisons**: relations expressing similarities or differences of more than one entity. Usually expressing an ordering.
  - E.g., “car x is cheaper than car y.”
  - Objective or subjective.
Typical opinion search queries

- Find the opinion of a person or organization (opinion holder) on a particular entity or an aspect of the entity.
  - E.g., what is Bill Clinton’s opinion on abortion?

- Find positive and/or negative opinions on a particular entity (or some aspects of the entity), e.g.,
  - customer opinions on a digital camera.
  - public opinions on a political topic.

- Find how opinions on an entity change over time.

- How entity A compares with entity B?
  - Gmail vs. Hotmail
Find the opinion of a person on X

- In some cases, the general search engine can handle it, i.e., using suitable keywords.
  - Bill Clinton’s opinion on abortion

**Reason:**

- One person or organization usually has only one opinion on a particular topic.
- The opinion is likely contained in a single document.
- Thus, a good keyword query may be sufficient.
Roadmap

- Introduction
- Opinion mining – problem definition
- Document level sentiment classification
- Aspect-based opinion mining
“I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. Although the battery life was not long, that is ok for me. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, and wanted me to return it to the shop. …”

What do we see?

- Opinions, targets of opinions, and opinion holders
Opinion mining – the abstraction
(Hu and Liu, KDD-04; Liu, Web Data Mining book 2007)

- Basic components of an opinion
  - Opinion holder: The person or organization that holds a specific opinion on a particular entity.
  - Entity: on which an opinion is expressed
  - Opinion: a view, attitude, or appraisal on an entity from an opinion holder.

- Objectives of opinion mining: many ...

- Let us abstract the problem
  - put existing research into a common framework

- We use consumer reviews of products to develop the ideas. Other opinionated contexts are similar.
Definition (entity): An entity $e$ is a product, person, event, organization, or topic. It is represented as
- a hierarchy of components, sub-components, and so on.
- Each node represents a component and is associated with a set of attributes of the component.

An opinion can be expressed on any node or attribute of the node.
To simplify our discussion, we use the term aspect (features) to represent both components & attributes.
Model of an entity

- An entity $e_i$ is represented with a finite set of aspects, $A = \{a_1, a_2, \ldots, a_n\}$.
- The entity can be expressed with any one of a final set of entity expressions $EE = \{ee_1, ee_2, \ldots, ee_s\}$.
- Each aspect $a \in A_i$ of the entity can be expressed with any one of a finite set of aspect expressions $AE = \{ae_1, ae_2, \ldots, ae_m\}$.
Model of a review

- An opinion holder $j$ comments on a subset of the aspects $S \subseteq A$ of an entity $e$.

- For each aspect $a \in S$ that $j$ comments on, he/she
  - chooses a word or phrase from $EE$ to describe the entity, and
  - chooses a word or phrase from $AE$ to describe the aspect, and
  - expresses a positive, negative or neutral opinion on $a$. 
An opinion is a quintuple

\[(e_j, a_{jk}, s_{ijkl}, h_i, t_l)\],

where

- \(e_j\) is a target entity.
- \(a_{jk}\) is a aspect of the entity \(e_j\).
- \(s_{ijkl}\) is the sentiment value of the opinion of the opinion holder \(h_i\) on aspect \(a_{jk}\) of entity \(e_j\) at time \(t_l\). \(s_{ijkl}\) is +ve, -ve, or neu, or a more granular rating.
- \(h_i\) is an opinion holder.
- \(t_l\) is the time when the opinion is expressed.
**Objective** — structure the unstructured

- **Objective**: Given an opinionated document,
  - Discover all quintuples \((e_j, a_k, s_{ijkl}, h_i, t_l)\), i.e., mine the five corresponding pieces of information in each quintuple,
  - Or, solve some simpler problems

- **With the quintuples,**
  - **Unstructured Text → Structured Data**
    - Traditional data and visualization tools can be used to slice, dice and visualize the results in all kinds of ways
    - Enable qualitative and quantitative analysis.
Sentiment Classification: document-level (Pang and Lee, 2008)

- **Classify a document** (e.g., a review) based on the overall sentiment expressed by opinion holder
  - **Classes**: Positive, or negative

- **Assumption**: each document focuses on a single object and contains opinions from a single op. holder.

- **E.g., thumbs-up or thumbs-down?**
  - “I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. Although the battery life was not long, that is ok for me. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, and wanted me to return it to the shop. …”
Sentence-level sentiment analysis has two tasks:

- **Subjectivity classification**: Subjective or objective.
  - **Objective**: e.g., *I bought an iPhone a few days ago.*
  - **Subjective**: e.g., *It is such a nice phone.*

- **Sentiment classification**: For subjective sentences or clauses, classify positive or negative.
  - **Positive**: *It is such a nice phone.*

**But**

- **subjective sentences ≠ +ve or –ve opinions**
  - E.g., *I think he came yesterday.*

- **Objective sentence ≠ no opinion**
  - Imply –ve opinion: *The phone broke in two days*
Sentiment classification at both document and sentence (or clause) levels are not enough,

- they do not tell what people like and/or dislike
- A positive opinion on an object does not mean that the opinion holder likes everything.
- An negative opinion on an object does not mean …..

Objective (recall): Discovering all quintuples

\[(o_j, f_{jk}, s_{ijkl}, h_i, t_l)\]

With all quintuples, all kinds of analyses become possible.
“I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. Although the battery life was not long, that is ok for me. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, and wanted me to return it to the shop. …”

Aspect Based Summary:

aspect1: Touch screen

Positive: 212
- The touch screen was really cool.
- The touch screen was so easy to use and can do amazing things.

Negative: 6
- The screen is easily scratched.
- I have a lot of difficulty in removing finger marks from the touch screen.

aspect2: battery life
Visual Comparison (Liu et al. WWW-2005)

Summary of reviews of Cell Phone 1

Comparison of reviews of Cell Phone 1 and Cell Phone 2

Voice  Screen  Battery  Size  Weight
Aspect-based opinion summary in Bing

The HP LaserJet 1020 Printer, an excellent laser printer for the cost-conscious consumer. It offers high-quality LaserJet printing in a compact size, and at a price you can afford. The reviews highlight its speed and reliability:

- **Speed**: 96%
- **Quality**: The quality is as good as any laserjet printer I've used and the speed is fast. Love Reading [www.amazon.com](http://www.amazon.com) 3/17/2006 more...
- **Ease of Use**: Quick and fast transaction. [Arthur L. Taylor](http://www.amazon.com) 2/5/2008 more...
- **Compatibility**: It's small and fast and very reliable. Muffinhead's mom [www.amazon.com](http://www.amazon.com) 1/9/2007 more...
Google Product Search

Sony Cyber-shot DSC-W370 14.1 MP Digital Camera (Silver)

$140 online, $170 nearby

Reviews

Summary - Based on 159 reviews

What people are saying

- pictures: "We use the product to take quickly photos."
- features: "Impressive panoramic feature."
- zoom/lens: "It also record better and focus better on sunny days."
- design: "It has the slightest grip but it's sufficient."
- video: "Video zoom is choppy."
- battery life: "Even better, the battery lasts long."
- screen: "I Love the Sony's 3" screen which I really wanted."
“This past Saturday, I bought a Nokia phone and my girlfriend bought a Motorola phone with Bluetooth. We called each other when we got home. The voice on my phone was not so clear, worse than my previous phone. The battery life was long. My girlfriend was quite happy with her phone. I wanted a phone with good sound quality. So my purchase was a real disappointment. I returned the phone yesterday.”
It is not Just ONE Problem

- \((e_j, a_k, s_{ijkl}, h_i, t_l})\)
  - \(e_j\) - a target entity: Named Entity Extraction (more)
  - \(a_{jk}\) - a aspect of \(e_j\): Information Extraction
  - \(s_{ijkl}\) is sentiment: Sentiment determination
  - \(h_i\) is an opinion holder: Information/Data Extraction
  - \(t_l\) is the time: Data Extraction

- Co-reference resolution
- Synonym match (voice = sound quality) …

- None of them is a solved problem!
Some commercial solutions give clients several example opinions in their reports.

Why not all? Accuracy could be the problem.

Accuracy: both

Precision: how accurate is the discovered opinions?

Recall: how much is left undiscovered?

Which sentence is better? (cordless phone review)

- (1) The voice quality is great.
- (2) I put the base in the kitchen, and I can hear clearly from the handset in the bedroom, which is very far.
Easier and Harder Problems

- Reviews are easier.
  - Objects/entities are given (almost), and little noise

- Forum discussions and blogs are harder.
  - Objects are not given, and a large amount of noise

- Determining sentiments seems to be easier.

- Determining objects and their features is harder.
  - Combining them is even harder.
Summary: opinion mining tasks

- **At the document (or review) level:**
  - **Task:** sentiment classification of reviews
    - **Classes:** positive, negative, and neutral
    - **Assumption:** each document (or review) focuses on a single entity (not true in many discussion posts) and contains opinion from a single opinion holder.

- **At the sentence level:**
  - **Task 1:** identifying subjective/opinionated sentences
    - **Classes:** objective and subjective (opinionated)
  - **Task 2:** sentiment classification of sentences
    - **Classes:** positive, negative and neutral.
    - **Assumption:** a sentence contains only one opinion (not true in many cases)
    - Next, we can also consider clauses or phrases.
At the aspect level:

- **Task 1** (entity extraction and grouping): Extract all entity expressions, and group synonymous entity expressions into entity clusters. Each cluster indicates a unique entity $e_i$.

- **Task 2** (aspect extraction and grouping): Extract all aspect expressions of the entities, and group synonymous aspect expressions into clusters. Each aspect expression cluster of entity $e_i$ indicates a unique aspect $a_{ij}$.

- **Task 3** (opinion holder and time extraction): Extract these pieces of information from the text or structured data.

- **Task 4** (aspect sentiment classification): Determine whether each opinion on an aspect is positive, negative or neutral.

- **Task 5** (opinion quintuple generation): Produce all opinion quintuples $(e_i, a_{ij}, o_{ijkl}, h_k, t_j)$ expressed in $D$. 
Roadmap

- Introduction
- Opinion mining – problem definition
- Document level sentiment classification
  - Supervised
  - Unsupervised
- Aspect-based opinion mining
Classify documents (e.g., reviews) based on the overall sentiments expressed by opinion holders (authors),

- Positive, negative, and (possibly) neutral
- Since in our model an entity \(e\) itself is also an aspect, then sentiment classification essentially determines the opinion expressed on \(e\) in each document (e.g., review).

Similar but different from topic-based text classification.

- In topic-based text classification, topic words are important.
- In sentiment classification, sentiment words are more important, e.g., great, excellent, horrible, bad, worst, etc.
Roadmap

- Introduction
- Opinion mining – problem definition
- Document level sentiment classification
  - Supervised
  - Unsupervised
- Aspect-based opinion mining
Machine Learning: Supervised Classification

Given:
- A description of an instance, \( d \in X \)
  - \( X \) is the instance language or instance space.
- A fixed set of classes: \( C = \{c_1, c_2, \ldots, c_J\} \)
- A training set \( D \) of labeled documents with each labeled document 
  \(<d,c> \in X \times C>\)

Determine:
- A learning method or algorithm which will enable us to learn a classifier \( \gamma: X \rightarrow C \)
- For a test document \( d \), assign it the class \( \gamma(d) \in C \)
Supervised review classification

- Machine learning, training on labeled data
- Represent document using attributes
- Most simple approach: „Bag of Words“ (word vector)

|     | a | and | aspect | be | better | ...
|-----|---|-----|--------|----|--------|------
| Text 1 | 6 | 4   | 0      | 1  | 2      | ...
| Text 2 | 5 | 1   | 1      | 0  | 3      | ...

- Needs only a tokenizer to split text into its tokens
I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet.
The bag of words representation

\[ Y(\gamma) = \begin{bmatrix}
great & 2 \\
love & 2 \\
recommend & 1 \\
laugh & 1 \\
happy & 1 \\
\ldots & \ldots \\
\end{bmatrix} \]
Supervised review classification

More linguistic features

- **Part-of-speech tagging (POS-Tagging)**
  - assigns each word its part of speech (= POS tag)
    - noun / adjective / adverb / determiner / ...
  - For word sense disambiguation
    - *Love*: noun vs. verb
  - To filter for specific word classes
    - Adjectives, and adverbs contain more sentiment than verbs

![Part-of-Speech:](image)

<table>
<thead>
<tr>
<th>PRP</th>
<th>RB</th>
<th>VBD</th>
<th>DT</th>
<th>RB</th>
<th>JJ</th>
<th>NNS</th>
<th>WRB</th>
<th>VBG</th>
<th>DT</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I</td>
<td>had</td>
<td>some</td>
<td>pretty</td>
<td>high</td>
<td>expectations</td>
<td>when</td>
<td>purchasing</td>
<td>the</td>
<td>GS3</td>
</tr>
</tbody>
</table>
POS examples

<table>
<thead>
<tr>
<th>Category</th>
<th>Part of Speech</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>noun</td>
<td>chair, bandwidth, pacing</td>
</tr>
<tr>
<td>V</td>
<td>verb</td>
<td>study, debate, munch</td>
</tr>
<tr>
<td>ADJ</td>
<td>adjective</td>
<td>purple, tall, ridiculous</td>
</tr>
<tr>
<td>ADV</td>
<td>adverb</td>
<td>unfortunately, slowly</td>
</tr>
<tr>
<td>P</td>
<td>preposition</td>
<td>of, by, to</td>
</tr>
<tr>
<td>PRO</td>
<td>pronoun</td>
<td>I, me, mine</td>
</tr>
<tr>
<td>DET</td>
<td>determiner</td>
<td>the, a, that, those</td>
</tr>
</tbody>
</table>
### Penn TreeBank POS Tagset

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>coordin. conjunction</td>
<td><em>and, but, or</em></td>
<td>SYM</td>
<td>symbol</td>
<td><em>+, %, &amp;</em></td>
</tr>
<tr>
<td>CD</td>
<td>cardinal number</td>
<td><em>one, two, three</em></td>
<td>TO</td>
<td>“to”</td>
<td><em>to</em></td>
</tr>
<tr>
<td>DT</td>
<td>determiner</td>
<td><em>a, the</em></td>
<td>UH</td>
<td>interjection</td>
<td><em>ah, oops</em></td>
</tr>
<tr>
<td>EX</td>
<td>existential ‘there’</td>
<td><em>there</em></td>
<td>VB</td>
<td>verb, base form</td>
<td><em>eat</em></td>
</tr>
<tr>
<td>FW</td>
<td>foreign word</td>
<td><em>mea culpa</em></td>
<td>VBD</td>
<td>verb, past tense</td>
<td><em>ate</em></td>
</tr>
<tr>
<td>IN</td>
<td>preposition/sub-conj</td>
<td><em>of, in, by</em></td>
<td>VBG</td>
<td>verb, gerund</td>
<td><em>eating</em></td>
</tr>
<tr>
<td>JJ</td>
<td>adjective</td>
<td><em>yellow</em></td>
<td>VBN</td>
<td>verb, past participle</td>
<td><em>eaten</em></td>
</tr>
<tr>
<td>JJR</td>
<td>adj., comparative</td>
<td><em>bigger</em></td>
<td>VBP</td>
<td>verb, non-3sg pres</td>
<td><em>eat</em></td>
</tr>
<tr>
<td>JJS</td>
<td>adj., superlative</td>
<td><em>wildest</em></td>
<td>VBZ</td>
<td>verb, 3sg pres</td>
<td><em>eats</em></td>
</tr>
<tr>
<td>LS</td>
<td>list item marker</td>
<td><em>1, 2, One</em></td>
<td>WDT</td>
<td>wh-determiner</td>
<td><em>which, that</em></td>
</tr>
<tr>
<td>MD</td>
<td>modal</td>
<td><em>can, should</em></td>
<td>WP</td>
<td>wh-pronoun</td>
<td><em>what, who</em></td>
</tr>
<tr>
<td>NN</td>
<td>noun, sing. or mass</td>
<td><em>llama</em></td>
<td>WP$</td>
<td>possessive wh-</td>
<td><em>whose</em></td>
</tr>
<tr>
<td>NNS</td>
<td>noun, plural</td>
<td><em>llamas</em></td>
<td>WRB</td>
<td>wh-adverb</td>
<td><em>how, where</em></td>
</tr>
<tr>
<td>NNP</td>
<td>proper noun, singular</td>
<td><em>IBM</em></td>
<td>$</td>
<td>dollar sign</td>
<td>$</td>
</tr>
<tr>
<td>NNPS</td>
<td>proper noun, plural</td>
<td><em>Carolin</em>as</td>
<td>#</td>
<td>pound sign</td>
<td>#</td>
</tr>
<tr>
<td>PDT</td>
<td>predeterminer</td>
<td><em>all, both</em></td>
<td>“</td>
<td>left quote</td>
<td>‘ or “</td>
</tr>
<tr>
<td>POS</td>
<td>possessive ending</td>
<td><em>’s</em></td>
<td>”</td>
<td>right quote</td>
<td>’ or ”</td>
</tr>
<tr>
<td>PRP</td>
<td>personal pronoun</td>
<td><em>I, you, he</em></td>
<td>(</td>
<td>left parenthesis</td>
<td>[ , {, &lt;</td>
</tr>
<tr>
<td>PRPS</td>
<td>possessive pronoun</td>
<td><em>your, one’s</em></td>
<td>)</td>
<td>right parenthesis</td>
<td>], } ) , &gt;</td>
</tr>
<tr>
<td>RB</td>
<td>adverb</td>
<td><em>quickly, never</em></td>
<td>,</td>
<td>comma</td>
<td>,</td>
</tr>
<tr>
<td>RBR</td>
<td>adverb, comparative</td>
<td><em>faster</em></td>
<td>:</td>
<td>sentence-final punctuation</td>
<td>. ! ?</td>
</tr>
<tr>
<td>RBS</td>
<td>adverb, superlative</td>
<td><em>fastest</em></td>
<td>;</td>
<td>mid-sentence punctuation</td>
<td>; ... - -</td>
</tr>
</tbody>
</table>

Source: Jurafsky & Martin (2009)
POS-Tagging

- POS-Tagging is not just a lexicon lookup, because
  - open word classes
  - ambiguous words

- Common approach of a POS-Tagger:
  - Lexicon-lookup for closed class words
  - Heuristics (e.g. based on suffix) for open class words
  - Tag of a term is dependent of its context
  - Tagger learns probabilities of tags for a term and tag sequences from (manually) annotated text
  - Tagger chooses tag sequence with highest probability
Supervised review classification

Two common applications for POS Tagging in Sentiment Analysis

- **Filtering:** Use adjectives and adverbs only, as they usually bear more sentiment compared to verbs, pronouns etc.

|        | apparently | absent | beautiful | bright | crappy | ...
|--------|-------------|--------|-----------|--------|--------|-----
| Text 1 | 2           | 4      | 0         | 1      | 2      | ... |
| Text 2 | 1           | 1      | 1         | 0      | 1      | ... |

- **Add POS-tags to each token for word sense disambiguation** *(to love vs. the love)*

|        | love_VBP | love_NN | the_Det | tell_VB | ...
|--------|----------|---------|---------|---------|-----
| Text 1 | 2        | 4       | 0       | 2       | ... |
| Text 2 | 1        | 1       | 1       | 1       | ... |
Further linguistic tools

- **Stemming**
  - Assigning each word its stem
  - *walking* -> *walk*
  - *fishing, fisher* -> *fish*
  - *meetings* -> *meet*
  - *good* -> *good*
  - *better* -> *better*

- **Lemmatization**
  - Assigning each word its lemma
  - *walking* -> *walk*
  - *(the) meeting* -> *meeting*
  - *(to be) meeting* -> *meet*
  - *better* -> *good*

A word has a single stem, namely the part of the word that is common to all its inflected variants.
["Analyzing Grammar", Paul Kröger, 2005]

A lemma (plural lemmas or lemmata) is the canonical form, dictionary form, or citation form of a set of words (headword).
["Lemma", Wikipedia, 8.3.2013]

---

**In general, stemming and lemmatization might help with various NLP tasks (especially for sparse data)**
Lemmatization

- Reduce inflectional/variant forms to base form

- E.g.,
  - *am, are, is* → *be*
  - *car, cars, car's, cars'* → *car*

- *the boy's cars are different colors* → *the boy car be different color*

- Lemmatization implies doing “proper” reduction to *dictionary headword form (the lemma)*
Stemming

- Reduce terms to their “roots” before indexing
- “Stemming” suggest crude affix chopping
  - language dependent
  - e.g., automate(s), automatic, automation all reduced to automat.

*for example compressed and compression are both accepted as equivalent to compress.*

*for example compress and compress are both accepted as equivalent to compress.*
Supervised Learning

- Learning can be performed with any supervised algorithm (e.g., as found in ML toolkits like RapidMiner)

- Classify documents as positive, negative or neutral
  - training for this class does not work well
  - better: train on positive and negative documents only
  - if prediction by classifier is not dominant for one of those classes, label document as neutral
Roadmap

- Introduction
- Opinion mining – problem definition
- Document level sentiment classification
  - Supervised
  - Unsupervised
- Aspect-based opinion mining
Unsupervised review classification: Lexicon-based

- Typically relies on a polarity lexicon listing positive and negative terms
  - Can include shifters \((\text{good vs. not good})\)
  - Can include intensifiers / diminishers \((\text{very nice})\)

- Example: SentiWordNet
  - A lexical resource in which each synset of WordNet, a widely-used lexical database of English, is associated to three numerical scores for objective, positive and negative valence
  - Example: the synset estimable\(\text{#a#3}\), corresponding to the sense of “may be computed or estimated” of the adjective estimable, has scores \(\text{Obj}(s)=1.0, \text{Pos}(s)=0.0\) and \(\text{Neg}(s)=0.0\) (i.e., total objective valence)
  - The synset estunable\(\text{#a#1}\) corresponding to the sense of “deserving of respect or higher regard” has scores \(\text{Obj}(s)=0.25, \text{Pos}(s)=0.75\) and \(\text{Neg}(s)=0.0\) (i.e., predominantly positive)
SentiWordNet

Figure 1: The graphical representation adopted by SentiWordNet for representing the opinion-related properties of a term sense.

Figure 2: SentiWordNet visualization of the opinion-related properties of the term estimable.

Adjective
3 senses found.

- **estimable(1)**
  - deserving of respect or high regard
  - \( P = 0.75, N = 0, O = 0.25 \)

- **honorable(5) good(4) respectable(2) estimable(2)**
  - deserving of esteem and respect; “all respectable companies give guarantees”;
    “ruined the family’s good name”
  - \( P = 0.625, N = 0.25, O = 0.125 \)

- **computable(1) estimable(3)**
  - may be computed or estimated: “a calculable risk”; “computable odds”; “estimable assets”
  - \( P = 0, N = 0, O = 1 \)
### SentiWordNet

#### Verb
2 senses found.

<table>
<thead>
<tr>
<th>Sense</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>short(1)</td>
<td>cheat someone by not returning him enough money</td>
</tr>
</tbody>
</table>

#### Adjective
15 senses found.

<table>
<thead>
<tr>
<th>Sense</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>short(1)</td>
<td>primarily temporal sense; indicating or being or seeming to be limited in duration; &quot;a short life&quot;; &quot;a short flight&quot;; &quot;a short holiday&quot;; &quot;a short story&quot;; &quot;only a few short months&quot;</td>
</tr>
<tr>
<td>short(2)</td>
<td>primarily spatial sense; having little length or lacking in length; &quot;short skirts&quot;; &quot;short hair&quot;; &quot;the board was a foot short&quot;; &quot;a short toss&quot;</td>
</tr>
<tr>
<td>short(3)</td>
<td>low in stature; not tall; &quot;his was short and stocky&quot;; &quot;short in stature&quot;; &quot;a short smokestack&quot;</td>
</tr>
<tr>
<td>inadequate(2) poor(7) short(4)</td>
<td>not sufficient to meet a need; &quot;an inadequate income&quot;; &quot;a poor salary&quot;; &quot;money is short&quot;; &quot;on short rations&quot;; &quot;food is in short supply&quot;; &quot;short on experience&quot;</td>
</tr>
</tbody>
</table>

Figure 3: SentiWordNet visualization of the opinion-related properties of the term short.
Polarity Lexicons

- SentiWordNet (Baccianella, Esuli, and Sebastiani 2010)
  - [http://sentiwordnet.isti.cnr.it/](http://sentiwordnet.isti.cnr.it/)

- Opinion Lexicon (Bing Liu)
  - [http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon](http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon)

- MPQA Subjectivity Lexicon (Wiebe, Wilson, and Cardie 2005)
  - [http://mpqa.cs.pitt.edu/](http://mpqa.cs.pitt.edu/)

  - [http://www.wjh.harvard.edu/~inquirer/](http://www.wjh.harvard.edu/~inquirer/)

- Linguistic Inquiry and Word Counts (LIWC)
  - [http://www.liwc.net/](http://www.liwc.net/) (commercial)

- Turney adjective List (Maite Taboada)
  - On request
Unsupervised review classification (Turney, ACL-02)

- Data: reviews from epinions.com on automobiles, banks, movies, and travel destinations.

- Step 1:
  - Part-of-speech tagging
  - Extracting two consecutive words (two-word phrases, aka “bigrams”) from reviews if their tags conform to some given patterns, e.g., an adjective preceding a noun (A N)
Step 2: Estimate the semantic orientation (SO) of the extracted phrases

- Use Point-wise mutual information

\[
PMI(word_1, word_2) = \log_2 \left( \frac{P(word_1 \land word_2)}{P(word_1)P(word_2)} \right)
\]

Semantic orientation (SO):

\[
SO(phrase) = PMI(phrase, \text{“excellent”}) - PMI(phrase, \text{“poor”})
\]

- Using AltaVista near operator to do search to find the number of hits to compute PMI and SO.
Step 3: Compute the average SO of all phrases

classify the review as **recommended** if average SO is positive, **not recommended** otherwise.

Table 2. An example of the processing of a review that the author has classified as **recommended**.\(^6\)

<table>
<thead>
<tr>
<th>Extracted Phrase</th>
<th>Part-of-Speech Tags</th>
<th>Semantic Orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td>online experience</td>
<td>JJ NN</td>
<td>2.253</td>
</tr>
<tr>
<td>low fees</td>
<td>JJ NNS</td>
<td>0.333</td>
</tr>
<tr>
<td>local branch</td>
<td>JJ NN</td>
<td>0.421</td>
</tr>
<tr>
<td>small part</td>
<td>JJ NN</td>
<td>0.053</td>
</tr>
<tr>
<td>online service</td>
<td>JJ NN</td>
<td>2.780</td>
</tr>
<tr>
<td>printable version</td>
<td>JJ NN</td>
<td>-0.705</td>
</tr>
<tr>
<td>direct deposit</td>
<td>JJ NN</td>
<td>1.288</td>
</tr>
<tr>
<td>well other</td>
<td>RB JJ</td>
<td>0.237</td>
</tr>
<tr>
<td>inconveniently located</td>
<td>RB VBN</td>
<td>-1.541</td>
</tr>
<tr>
<td>other bank</td>
<td>JJ NN</td>
<td>-0.850</td>
</tr>
<tr>
<td>true service</td>
<td>JJ NN</td>
<td>-0.732</td>
</tr>
<tr>
<td><strong>Average Semantic Orientation</strong></td>
<td></td>
<td><strong>0.322</strong></td>
</tr>
</tbody>
</table>

Table 3. An example of the processing of a review that the author has classified as **not recommended**.

<table>
<thead>
<tr>
<th>Extracted Phrase</th>
<th>Part-of-Speech Tags</th>
<th>Semantic Orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td>little difference</td>
<td>JJ NN</td>
<td>-1.615</td>
</tr>
<tr>
<td>clever tricks</td>
<td>JJ NNS</td>
<td>-0.040</td>
</tr>
<tr>
<td>programs such</td>
<td>NNS JJ</td>
<td>0.117</td>
</tr>
<tr>
<td>possible moment</td>
<td>JJ NN</td>
<td>-0.668</td>
</tr>
<tr>
<td>unethical practices</td>
<td>JJ NNS</td>
<td>-8.484</td>
</tr>
<tr>
<td>low funds</td>
<td>JJ NNS</td>
<td>-6.843</td>
</tr>
<tr>
<td>old man</td>
<td>JJ NN</td>
<td>-2.566</td>
</tr>
<tr>
<td>other problems</td>
<td>JJ NNS</td>
<td>-2.748</td>
</tr>
<tr>
<td>probably wondering</td>
<td>RB VBG</td>
<td>-1.830</td>
</tr>
<tr>
<td>virtual monopoly</td>
<td>JJ NN</td>
<td>-2.050</td>
</tr>
<tr>
<td>other bank</td>
<td>JJ NN</td>
<td>-0.850</td>
</tr>
<tr>
<td>extra day</td>
<td>JJ NN</td>
<td>-0.286</td>
</tr>
<tr>
<td>direct deposits</td>
<td>JJ NNS</td>
<td>5.771</td>
</tr>
<tr>
<td>online web</td>
<td>JJ NN</td>
<td>1.936</td>
</tr>
<tr>
<td>cool thing</td>
<td>JJ NN</td>
<td>0.395</td>
</tr>
<tr>
<td>very handy</td>
<td>RB JJ</td>
<td>1.349</td>
</tr>
<tr>
<td>lesser evil</td>
<td>RBR JJ</td>
<td>-2.288</td>
</tr>
<tr>
<td><strong>Average Semantic Orientation</strong></td>
<td></td>
<td><strong>-1.218</strong></td>
</tr>
</tbody>
</table>
Roadmap

- Introduction
- Opinion mining – problem definition
- Document level sentiment classification
- Aspect-based opinion mining

... Will be continued in Web Content Mining - Part 2 - ...
Web Mining

Web Content Mining
- Part 2 -

Simone Paolo Ponzetto
Stefano Faralli

FSS 2017
Roadmap

- Part 1 -
- Introduction
- Opinion mining – problem definition
- Document level sentiment classification

- Part 2 -
- Information Extraction
- Aspect-based opinion mining
- Part 2 -

- Information Extraction

- Aspect-based opinion mining
Sentiment Analysis / Opinion Mining

The basic task in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature/aspect level.

- **Polarity Values**
  - Positive, neutral, negative
  - Likert scale (1 to 10)

- **Application Examples**
  - **Document-Level**
    - tweet analysis about politicians
  - **Feature/Aspect-Level**
    - analysis of product reviews
An opinion is a quintuple

\[(e_j, a_{jk}, s_{ijkl}, h_i, t_l),\]

where

- \(e_j\) is a target entity.
- \(a_{jk}\) is a aspect of the entity \(e_j\).
- \(s_{ijkl}\) is the sentiment value of the opinion of the opinion holder \(h_i\) on aspect \(a_{jk}\) of entity \(e_j\) at time \(t_l\). \(s_{ijkl}\) is +ve, -ve, or neu, or a more granular rating.
- \(h_i\) is an opinion holder.
- \(t_l\) is the time when the opinion is expressed.
Objective – structure the unstructured

Objective: Given an opinionated document,
- Discover all quintuples \((e_i, a_k, so_{ijkl}, h_i, t_l)\),
  - i.e., mine the five corresponding pieces of information in each quintuple,
- Or, solve some simpler problems

With the quintuples,
- Unstructured Text \rightarrow Structured Data
  - Traditional data and visualization tools can be used to slice, dice and visualize the results in all kinds of ways
  - Enable qualitative and quantitative analysis.
Sentiment Classification: document-level (Pang and Lee, 2008)

- Classify a document (e.g., a review) based on the overall sentiment expressed by opinion holder
  - Classes: Positive, or negative

- Assumption: each document focuses on a single object and contains opinions from a single op. holder.

- E.g., thumbs-up or thumbs-down?
  - “I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. Although the battery life was not long, that is ok for me. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, and wanted me to return it to the shop. …”
Subjectivity Analysis: sentence-level (Wiebe et al 2004)

- Sentence-level sentiment analysis has two tasks:
  - **Subjectivity classification**: Subjective or objective.
    - **Objective**: e.g., *I bought an iPhone a few days ago.*
    - **Subjective**: e.g., *It is such a nice phone.*
  - **Sentiment classification**: For subjective sentences or clauses, classify positive or negative.
    - **Positive**: *It is such a nice phone.*

- **But**
  - Subjective sentences ≠ +ve or –ve opinions
    - E.g., *I think he came yesterday.*
  - Objective sentence ≠ no opinion
    - Imply –ve opinion: *The phone broke in two days*
Aspect-Based Sentiment Analysis

- Sentiment classification at both document and sentence (or clause) levels are not enough,
  - they do not tell what people like and/or dislike
  - A positive opinion on an object does not mean that the opinion holder likes everything.
  - An negative opinion on an object does not mean …..

- Objective (recall): Discovering all quintuples
  \((o_j, f_{jk}, s_{ijkl}, h_i, t_l)\)

- With all quintuples, all kinds of analyses become possible.
“I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. Although the battery life was not long, that is ok for me. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, and wanted me to return it to the shop. …”

Aspect-Based Opinion Summary (Hu & Liu, KDD-2004)

Aspect Based Summary:

aspect1: Touch screen
Positive: 212
- The touch screen was really cool.
- The touch screen was so easy to use and can do amazing things.

Negative: 6
- The screen is easily scratched.
- I have a lot of difficulty in removing finger marks from the touch screen.

aspect2: battery life
Visual Comparison (Liu et al. WWW-2005)

Summary of reviews of

Cell Phone 1

+ Voice

Comparison of reviews of

Cell Phone 1

Cell Phone 2

+ Screen

+ Battery

+ Size

+ Weight
Opinion Mining is Hard!

“This past Saturday, I bought a Nokia phone and my girlfriend bought a Motorola phone with Bluetooth. We called each other when we got home. The voice on my phone was not so clear, worse than my previous phone. The battery life was long. My girlfriend was quite happy with her phone. I wanted a phone with good sound quality. So my purchase was a real disappointment. I returned the phone yesterday.”
It is not Just ONE Problem

\((e_j, a_k, so_{ijkl}, h_i, t_l)\),

- \(e_j\) - a target entity: Named Entity Extraction (more)
- \(a_{jk}\) - a aspect of \(e_j\): Information Extraction
- \(so_{ijkl}\) is sentiment: Sentiment determination
- \(h_i\) is an opinion holder: Information/Data Extraction
- \(t_l\) is the time: Data Extraction

- Co-reference resolution
- Synonym match (voice = sound quality) …

- None of them is a solved problem!
Information extraction (IE) is the task of automatically extracting structured information from unstructured or semi-structured machine-readable documents.

An umbrella term for a variety of heterogeneous tasks:

- Named Entity Recognition
- Relation extraction
- ...

Definition
Identify mentions of entities in text

Classify them into a predefined set of categories of interest:

- Person Names: Prof. Jerry Hobbs, Jerry Hobbs
- Organizations: Hobbs corporation, FbK
- Locations: Ohio
- Date and time expressions: February 2010
- E-mail: mkg@gmail.com
- Web address: www.usc.edu
- Names of drugs: paracetamol
- Names of ships: Queen Marry
- Bibliographic references:
- ...
## NER: sample classes

<table>
<thead>
<tr>
<th>Type</th>
<th>Tag</th>
<th>Sample Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>People</td>
<td>PER</td>
<td>Individuals, fictional characters, small groups</td>
</tr>
<tr>
<td>Organization</td>
<td>ORG</td>
<td>Companies, agencies, political parties, religious groups, sports teams</td>
</tr>
<tr>
<td>Location</td>
<td>LOC</td>
<td>Physical extents, mountains, lakes, seas</td>
</tr>
<tr>
<td>Geo-Political Entity</td>
<td>GPE</td>
<td>Countries, states, provinces, counties</td>
</tr>
<tr>
<td>Facility</td>
<td>FAC</td>
<td>Bridges, buildings, airports</td>
</tr>
<tr>
<td>Vehicles</td>
<td>VEH</td>
<td>Planes, trains, and automobiles</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>People</td>
<td><em>Turing</em> is often considered to be the father of modern computer science.</td>
</tr>
<tr>
<td>Organization</td>
<td>The <em>IPCC</em> said it is likely that future tropical cyclones will become more intense.</td>
</tr>
<tr>
<td>Location</td>
<td>The <em>Mt. Sanitas</em> loop hike begins at the base of <em>Sunshine Canyon</em>.</td>
</tr>
<tr>
<td>Geo-Political Entity</td>
<td><em>Palo Alto</em> is looking at raising the fees for parking in the University Avenue district.</td>
</tr>
<tr>
<td>Facility</td>
<td>Drivers were advised to consider either the <em>Tappan Zee Bridge</em> or the <em>Lincoln Tunnel</em>.</td>
</tr>
<tr>
<td>Vehicles</td>
<td>The updated <em>Mini Cooper</em> retains its charm and agility.</td>
</tr>
</tbody>
</table>

Source: Jurafsky & Martin (2009)
Rule Based NER

- Create regular expressions to extract:
  - Telephone number
  - E-mail
  - Capitalized names

- Example: locations
  - Capitalized word + {street, boulevard, avenue}
    
    Ex. *Fifth avenue*
  - Capitalized word + {city, center, river}
    
    Ex. *New York city*
    
    *Hudson river*
Why simple things do not work?

- Capitalization is a strong indicator for capturing proper names

- But it still can be tricky:
  - first word of a sentence is capitalized
  - sometimes titles in web pages are all capitalized
  - nested named entities contain non-capital words
    - University of Southern California is Organization
  - all nouns in German are capitalized
Why simple things do not work?

- No gazetteer contains all existing proper names
- New proper names constantly emerge, e.g., *movie titles, books, singers, restaurants, etc.*

- Proper names are ambiguous
  - Jordan the *person* vs. Jordan the *location*
  - JFK the *person* vs. JFK the *airport*
  - May the *person* vs. May the *month*

- Machine learning methods offer a possible solution to these problems...
Machine Learning based NER

- **NED**: Identify named entities using BIO tags
  - **B**: beginning of an entity
  - **I**: continues the entity
  - **O**: word outside the entity

Adam_B Smith_I works_O for_O IBM_B ,_O London_B ._O
Machine Learning based NER

- **NED**: Identify named entities using BIO tags
  - B beginning of an entity
  - I continues the entity
  - O word outside the entity

- **NEC**: Classify into a predefined set of categories
  - Person names
  - Organizations (companies, governmental organizations, etc.)
  - Locations (cities, countries, etc.)
  - Miscellaneous (movie titles, sport events, etc.)
Machine Learning based NER

Source: Jurafsky & Martin (2009)
**k Nearest Neighbors**

- Learning is just storing the representations of the training examples.

- Given a test instance $x$:
  - compute similarity between $x$ and all training examples
  - collect the categories among $x$’s $k$ nearest neighbors
  - assign $x$ the same category as most of its similar examples in the training set
1-Nearest Neighbor
3-Nearest Neighbors

choose the category of the closest neighbor (can be erroneous due to noise)

choose the category of the majority of the neighbors
5-Nearest Neighbors

The value of $k$ is typically odd to avoid ties
**k Nearest Neighbors: issues**

- **Pros**
  - + robust
  - + simple
  - + training is very fast (storing examples)

- **Cons**
  - - depends on similarity measure & k-NNs
  - - easily fooled by irrelevant attributes
Decision Trees

- The classifier has a tree structure, where each node is either:
  - a leaf node which indicates the value of the target attribute (class)
  - a decision node which specifies some test to be carried out on an attribute-value, with one branch and sub-tree for each possible outcome of the test

- A test instance is classified by starting at the root of the tree and moving through it until a leaf node is reached, which provides the class for the instance
Building Decision Trees

- We can learn decision trees from labeled data.
- This is typically achieved with a top-down, greedy search.
- At each step we select the best attribute, i.e., the most useful for classifying examples using information theoretic measures like, e.g., Information Gain or Gain Ratio.

Each internal node tests an attribute.

Each branch corresponds to an attribute value node.

Each leaf node assigns a classification.

```
isCapitalized
  1
  0
  NO

isPersonName
  1
  0
  NO

isLiving
  1
  0
  NO

YES  NO
```
Decision trees: issues

- **Pros**
  - generate understandable rules
  - gives a clear features ranking

- **Cons**
  - error prone in multi-class classification and small datasets
Features for NE Detection

- **Orthographic:**
  - *initial-caps*
  - *roman-number*
  - *Acronym*
  - *all-caps*
  - *contains-dots*
  - *punctuation-mark*
  - *all-digits*
  - *contains-hyphen*
  - *URL*

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

- **Left Predictions:** the tag predicted for $w_{-1}$, $w_{-2}$, …

- **Part-of-speech tag**
Features for NE Classification

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

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

- **Bag-of-Words**
  - words in $[-5,\ldots,+5]$ window

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

- **Gazetteers**
  - geographical
  - first name
  - surname
  - company names
What is Relation Extraction?

- The automatic extraction of *structured* semantic relations from *unstructured* documents.
- A limited form of natural language understanding

**Example**

Systems find instances of target relations.

*Example:*

\[ \text{HeadquarteredIn}(\text{company}, \text{city}) \]

Some newswire text:

EMI Music Publishing Latin America, the Latin music and entertainment arm of the EMI music conglomerate, has its headquarters in Miami, FL.

\[ \text{HeadquarteredIn}(\text{EMI}, \text{Miami}) \]
Extracting Corporate Information

Source web page. Color highlights indicate type of information. (e.g., red = name)

<table>
<thead>
<tr>
<th>People/Titles</th>
<th>Addresses</th>
<th>Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greg Erman -- President &amp; CEO</td>
<td>10 Maguire Road, Suite 330, Lexington MA 02421-3112</td>
<td>CEO Marri</td>
</tr>
<tr>
<td>Marcia J. Hooper -- Partner</td>
<td>Ten Maguire Road, Suite 33</td>
<td>Capital Pa Internation</td>
</tr>
<tr>
<td>John Losier -- President and CEO</td>
<td>10 Maguire Road, Lexington, MA 02421</td>
<td>President Software Capiton</td>
</tr>
<tr>
<td>Robert C. Fleming -- Principal</td>
<td>100 Fifth Avenue, New York, NY 10011-6901</td>
<td>Burlington-</td>
</tr>
<tr>
<td>James C. Furnival -- Partner</td>
<td></td>
<td>Digital Equip</td>
</tr>
</tbody>
</table>

Source: Whizbang! Labs
Harvesting Product Information
Example: a book description from Amazon

....
</td></tr>
</table>
<b class="sans">The Age of Spiritual Machines : When Computers Exceed Human Intelligence</b><br>
by <a href="/exec/obidos/search-handle-url/index=books&field-author=Kurzweil%2C%20Ray/002-6235079-4593641">Ray Kurzweil</a><br>
</font><br>
<a href="http://images.amazon.com/images/P/0140282025.01.LZZZZZZZ.jpg">
<img src="http://images.amazon.com/images/P/0140282025.01.MZZZZZZZ.gif" width=90 height=140 align=left border=0></a>
<font face=verdana,arial,helvetica size=-1>
<span class="small">
<b>List Price:</b> $14.95<br>
<b>Our Price: $11.96</b><br>
<b>You Save: $2.99 (20%)</b>
</span>
<p>...</p>
Example: a book description from Amazon

- **Title:** The Age of Spiritual Machines: When Computers Exceed Human Intelligence
- **Author:** Ray Kurzweil
- **List-Price:** $14.95
- **Price:** $11.96
Extraction Patterns based on RegExp

- Specify an item to extract for a slot using a regular expression.
  - Price pattern: “\b\$\d+(\.\d{2})?\b”

- Can be extended to use also a preceding (pre-filler) and succeeding (post-filler) pattern to identify a proper context.
  - Pre-filler pattern: “<b>List Price:</b> <span class=listprice>”
  - Filler pattern: “\$\d+(\.\d{2})?\b”
  - Post-filler pattern: “</span>”

- Patterns of this kind can also be learned using machine learning methods
NLP-based Relation Extraction

- Simple regex patterns work best with automatically generated web pages.

- When extracting from unstructured, human-written text, we need to rely instead on textual (e.g., morpho-syntactic) patterns.

Example: instances of a class

- $\text{NP}_1$ “such as” $\text{NPList}_2$
- $\text{NP}_2$ “and other” $\text{NP}_2$
- $\text{NP}_1$ “is a” $\text{NP}_2$

“Compact system cameras such as the Sony NEX-7 ...”

„The Panasonic Lumix and other compact system cameras...“
Standard techniques for NLP-based IE

1. Manually constructed patterns
2. Supervised classifiers
3. Pattern-learning and bootstrapping
Supervised IE using classifiers as extractors

Raw Data → Labeled Training Data → Learning Algorithm

Extractor

- **Kirkland**-based **Microsoft** is the largest software company.
- **Boeing** moved its headquarters to **Chicago** in 2003.
- Hank Levy was named chair of Computer Science & Engr.

... 

HeadquarterOf(<company>,<city>)
Bootstrapping

Source: Jurafsky & Martin (2009)
Bootstrapping: example

Seed Examples
- The name of the rose – U. Eco
- Born to run – C. McDougall

Rule Learning

Extraction Rules
- X is author of Y
- X, author of Y
- X wrote Y

High-confidence Extractions
Bootstrapping: example

Seed Examples

<table>
<thead>
<tr>
<th>Example</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>The name of the rose – U. Eco</td>
<td></td>
</tr>
<tr>
<td>Born to run – C. McDougall</td>
<td></td>
</tr>
<tr>
<td>Madame Bovary – G. Flaubert</td>
<td></td>
</tr>
<tr>
<td>Catcher in the rye – J.D. Salinger</td>
<td></td>
</tr>
</tbody>
</table>

Rule Learning

Extraction Rules

<table>
<thead>
<tr>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>X is author of Y</td>
</tr>
<tr>
<td>X, author of Y</td>
</tr>
<tr>
<td>X wrote Y</td>
</tr>
<tr>
<td>The essayist X is the of bestselling author of Y</td>
</tr>
</tbody>
</table>
Roadmap

- Part 2 -

- Information Extraction
- Aspect-based opinion mining
http://xkcd.com/937/
Objective: find what reviewers (opinion holders) liked and disliked

- Product aspects and opinions on the aspects

Since the number of reviews on an entity can be large, an opinion summary should be produced.

- Desirable to be a structured summary.
- Easy to visualize and to compare.
- Analogous to but different from multi-document summarization.
Aspect-based opinion mining tasks

- **Task 1 (entity extraction and grouping):** Extract all entity expressions, and group synonymous entity expressions into entity clusters. Each cluster indicates a unique entity $e_i$.

- **Task 2 (aspect extraction and grouping):** Extract all aspect expressions of the entities, and group synonymous aspect expressions into clusters. Each aspect expression cluster of entity $e_i$ indicates a unique aspect $a_{ij}$.

- **Task 3 (opinion holder and time extraction):** Extract these pieces of information from the text or structured data.

- **Task 4 (aspect sentiment classification):** Determine whether each opinion on an aspect is positive, negative or neutral.

- **Task 5 (opinion quintuple generation):** Produce all opinion quintuples $(e_i, a_{ij}, oo_{ijkl}, h_k, t_l)$ expressed in $D$. 
The tasks

We have 5 tasks, but only focus on two.

- Task 2 (aspect extraction and grouping): Extract all aspect expressions of the entities, and group synonymous aspect expressions into clusters. Each aspect expression cluster of entity $e_i$ indicates a unique aspect $a_{ij}$.

- Task 4 (aspect sentiment classification): Determine whether each opinion on an aspect is positive, negative or neutral.
Frequent feature extraction (Hu & Liu, 2004)

Tasks

1. Identifying product features
2. Extracting opinion words for features
3. [Producing a summary of the given information]

Not discussed here
Frequent feature extraction (Hu & Liu, 2004)

1. Identifying product features

- **Intuition:**
  Important features are talked about by many customers

- **Approach**
  - reviews are POS-tagged
  - nouns and noun phrases are extracted as potential features
  - their occurrence frequencies are counted and only frequent ones are kept
  - threshold can be set experimentally
2a. Opinion words extraction

- POS-tags are used again

- For each frequent feature, the closest adjective is chosen as its opinion word

„The strap is horrible and gets in the way...“

„The horrible strap is attached to the camera...“

- In both examples, horrible will be the opinion word for strap.
2.b Determine semantic orientation of opinion term

- Polarity of opinion term could be looked up in sentiment lexicon

- In the work of Hu & Liu, as a lexicon was not available, they looked for adjectives with known polarities in WordNet that are related to their extracted ones
OPINE (Popescu & Etzioni, 2005)

Opine’s subtasks

- I. Identify product features.
- II. Identify opinions regarding product features.
- III. Determine the polarity of opinions.
- (IV. Rank opinions based on their strength.)

Not discussed here
Opine

Review Summary

Service quality: excellent (3), good (2), best, professional, better, view all (8)

Service attention: attentive (2)

Room beauty: absolutely beautiful, beautiful, view all (2)

User comments:

The service was excellent and our room was absolutely beautiful. Read more

When compared to Mandarin Oriental New York, Room beauty is

- worse at The Premier (33 others)

Quality: best, finest, love, better, view all (4)

Staff courtesy: extremely courteous, courteous, view all (2)

Beauty: beautiful

Room quality: gorgeous, complementary, view all (2)

Food quality: lovely, nice, view all (2)

Service discretion: discreet
**Opine: Feature Extraction**

- **Product classes**
  - *Hotels*

- **Instances**
  - *Trump International*

<table>
<thead>
<tr>
<th>Extracted Features</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Properties</td>
<td>Quality Size</td>
</tr>
<tr>
<td>Parts</td>
<td>Room</td>
</tr>
<tr>
<td>Features of parts</td>
<td>RoomSize</td>
</tr>
<tr>
<td>Related concepts</td>
<td>Neighborhood</td>
</tr>
<tr>
<td>Features of related concepts</td>
<td>NeighborhoodSafety</td>
</tr>
</tbody>
</table>

**OPINE also extracts opinion phrases**
Opine: Feature Extraction

I loved the hot water and the clean bathroom.

The fan was broken and our room was hot the entire time.

I like a nice, hot room when the snow piles up outside.

Extract noun phrases np such that np contains only nouns and frequency(np)>1 as potential features.
I loved the hot water and the clean bathroom.

The fan was broken and our room was hot the entire time.

I like a nice, hot room when the snow piles up outside.

Assess potential features using bootstrapped lexical patterns (*discriminators*)

Examples
- X of Y
- Y has X
- Y’s X
- Y with X
- Y comes with X
- Y equipped with X
- Y contains X
- Y offers X
I loved the hot water and the clean bathroom.

The fan was broken and our room was hot the entire time.

I like a nice, hot room when the snow piles up outside.

Assess potential features using discriminators

\[
\text{PMI(hotel’s}[Y], \text{room}) = \frac{\text{hits(“hotel’s room”)}}{\text{hits(“hotel’s”)}} \times \text{hits(“room”)}
\]

\[
\text{PMI(hotel’s } [Y], \text{room}) = 0.54 \times 10^{-13}
\]

\[
\text{PMI(hotel’s } [Y], \text{snow}) = 0.64 \times 10^{-16}
\]

\[
\text{PMI(hotel’s } [Y], \text{room}) >> \text{PMI(hotel’s } [Y], \text{snow})
\]
I loved the hot water and the clean bathroom.

The fan was broken and our room was hot the entire time.

I like a nice, hot room when the snow piles up outside.

Assess potential features using discriminators

\[
\text{PMI(\text{hotel’s}[Y], \text{room}) = \frac{\text{hits(“hotel’s room”)}}{\text{hits(“hotel’s”)}} \times \text{hits(“room”)}}
\]

\[
\text{PMI(\text{hotel’s}[Y], \text{room})} = 0.54 \times 10^{-13}
\]

\[
\text{PMI(\text{hotel’s}[Y], \text{snow})} = 0.64 \times 10^{-16}
\]

\[
\text{PMI(\text{hotel’s}[Y, \text{room})} \gg \text{PMI(\text{hotel’s}[Y, \text{snow})}
\]
Given feature $f$, extract $po$ if:

$\exists po \text{ such that } pos(po) = adj|nn, \ mod(po,f)$

- $f = \text{feature}$
- $po = \text{potential opinion}$
- $pos = \text{part-of-speech tag}$
- $adj = \text{adjective}$
- $nn = \text{noun}$
- $mod = \text{modifies}$

„Given feature $f$, extract $po$ if

- there is a $po$ with its pos-tag being $adj$ or $nn$
- and $po$ modifies $f$

„I love its $great$ resolution“, „and need $fast$ autofocus“
Opine: Opinion Phrase Extraction

Given feature \( f \), extract \( po \) if:
\[
\exists po \text{ such that } pos(po) = adv, \exists (S, O) \text{ such that } S = f, O = po
\]

- \( f = \text{feature} \)
- \( po = \text{potential opinion} \)
- \( pos = \text{part-of-speech tag} \)
- \( adv = \text{adverb} \)
- \( S = \text{subject} \)
- \( O = \text{object} \)

„Given feature \( f \), extract \( po \) if“
- there is a \( po \) with its pos-tag being \( adv \)
- \( f \) is the \text{subject} and \( po \) is the \text{object}

„The scanner works well“ „the camera shoots fast“
Opine: Semantic Orientation

The room was hot(-) and stuffy(-).

After freezing for hours, the room was nice(+) and hot(+).

cold basic loud visible casual modern central quiet

- After the potential opinion phrases are extracted, OPINE assigns them one of 3 semantic orientation labels (positive, negative, neutral)

- More formally, OPINE computes a SO label for a word in the context of a product feature and a sentence. For example, hot is negative...

- Initial scores of phrases can be derived from a sentiment lexicon
Opine: Semantic Orientation

- Task: Compute the SO label for a (word, feature, sentence) tuple

- OPINE solves the task in 3 steps.
  - An overall SO label for a word is computed.
    - SO(word)
  - A SO label for a word in the context of a given feature is computed
    - SO(word, feature)
  - A SO label for a word in the context of a given feature and a given sentence is computed
    - SO(word, feature, sentence)

- Each solution step = labeling problem ➔ relaxation labeling
Opine: Relaxation Labeling

Unsupervised classification technique

Input

- Set of objects (e.g. words)
- Set of labels (e.g. SO labels)
- Initial probabilities for each object's possible labels
- Definition of an object's neighborhood (other objects)
- Definition of neighborhood features
- Definition of support function for object label

- Example: The word "nice" participates in conjunction "and" together with another word whose label is estimated positive.
Opine: Relaxation Labeling

- Used when the label of a given object is constrained by the labels of other objects (its neighborhood)
  - Here: label of a word is influenced by
    - other words attached to it in the sentence
    - by the known labels of synonymous words
    - ...

- The influence of an object’s neighborhood on the object’s label is quantified by a support function.

- Starts with an initial assignment of labels to objects and iteratively modifies this assignment.
  - At each iteration, it updates the probability of each label of each object based on current probability and on the current labels of the object’s neighbors.

- RL stops when some termination criterion is met (e.g. when global label assignment stays constant)
Building word neighborhoods:
- conjunctions, disjunctions
- syntactic attachment rules
- WordNet synonymy/antonymy
- morphology information

Opine: Relaxation Labeling