Web Mining

Web Content Mining
- Part 2 -

Simone Paolo Ponzetto
Stefano Faralli

FSS 2016
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
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.
- \(a_k\) 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 \(\rightarrow\) 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*
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

<table>
<thead>
<tr>
<th>Feature</th>
<th>Cell Phone 1</th>
<th>Cell Phone 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voice</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Screen</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>Battery</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Size</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Weight</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

Comparison of reviews of

- Cell Phone 1
- Cell Phone 2
“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!
Information Extraction

Definition

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
- ...

...
Named Entity Recognition and Classification

- 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:
  - ...

<PER>Prof. Jerry Hobbs</PER> taught CS544 during <DATE>February 2010</DATE>.<PER> Jerry Hobbs</PER> killed his daughter in <LOC>Ohio</LOC>. <ORG>Hobbs corporation</ORG> bought <ORG>FbK</ORG>. 
# 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.)

Adam_B-PER Smith_I-PER works_O for_O IBM_B-ORG ,_O London_B-LOC ._O
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.
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
  - lowercased
  - punctuation
  - mark
  - quote
  - other
  - 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 $[-3,...,+3]$ window

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

- **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**
  - 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.

E.g., `HeadquarteredIn(<company>, <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.

`HeadquarteredIn(EMI, Miami)`
Extracting Corporate Information

For more information contact:

Dan Carter  
Marketing Manager  
MarketSoft Corporation  
781-674-0000 x 302  
carter@marketsoft.com

Brent Skinner  
Account Coordinator  
The Weber Group  
617-520-7054  
bskinner@webergroup.com

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

Source: Whizbang! Labs

MarketSoft Corporation (?)

http://marketsoft.com

Street address: 10 Maguire Road, Suite 330
City: Lexington
State: MA
Zip code: 02421-3112
Telephone: 781-674-0000 (?)
Fax: (212) 924-0240 (???)
Email: info@marketsoft.com
IC code: 7372 [Prepackaged software]

People/Titles
Greg Erman -- President & CEO, MarketSoft
Maria J. Hooper -- Partner
John Losier -- President and CEO
Robert C. Fleming -- Principal
James C. Furnival -- Partner

Addresses
10 Maguire Road, Suite 330, Lexington MA 02421-3112
Ten Maguire Road, Suite 330, Lexington, MA, 02421
10 Maguire Road, Lexington, MA, 02421
104 Fifth Avenue, New York, NY 10011-5901

Companies
CEOMarketing
Capital Params
Interact
President (Software)
Burlington

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>
<font face=verdana,arial,helvetica size=-1>
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>
<br>
<font face=verdana,arial,helvetica size=-1>
<br>
<b>List Price:</b> $14.95<br>
<b>Our Price: </b>$11.96<br>
<b>You Save: </b>$2.99 (20%)</font><br>
</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
  - NP₁ “such as” NPl₁
  - NP₂ “and other” NP₂
  - NP₁ “is a” NP₂

  “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

Pattern-Based Relation Extraction

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
- The name of the rose – U. Eco
- Born to run – C. McDougall
- Madame Bovary – G. Flaubert
- Catcher in the rye – J.D. Salinger

Rule Learning

Extraction Rules
- X is author of Y
- X, author of Y
- X wrote Y
- The essayist X is the of bestselling author of Y

High-confidence Extractions
Roadmap

- Part 2 -

- Information Extraction
- Aspect-based opinion mining
http://xkcd.com/937/

APP STORE

TORNADO GUARD
FROM DROID CODER 2187

PLAYS A LOUD ALERT SOUND
WHEN THERE IS A TORNADO
WARNING FOR YOUR AREA.

RATING: ★★★★★
BASED ON 4 REVIEWS

USER REVIEWS:

★★★★★ GOOD UI!
MANY ALERT CHOICES.

★★★★★ RUNNING
GREAT, NO CRASHES

★★★★★ I LIKE HOW YOU
CAN SET MULTIPLE LOCATIONS

★★★☆☆ APP DID NOT
WARN ME ABOUT TORNADO.

THE PROBLEM WITH
AVERAGING STAR RATINGS
Aspect-based opinion mining and summarization

- **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 $D$. 
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
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

OPINE
Ana-Maria Popescu, Bao Nguyen, Oren Etzioni

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**
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

PMI(hotel’s[Y], room) =

hits(“hotel’s room”) / hits(“hotel’s”)*hits(“room”)

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

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

PMI(hotel’s [Y], room) >> PMI(hotel’s [Y],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

PMI(hotel’s[Y], room) =

\[
\text{hits(“hotel’s room”) / hits(“hotel’s”) \times \text{hits(“room”)}}
\]

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

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

PMI(hotel’s [Y], room) >> PMI(hotel’s [Y], snow)
Opine: Opinion Phrase Extraction

Given feature $f$, extract $po$ if:

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

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

„Given feature $f$, extract $po$ if"

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

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

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

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

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

- e.g. the word „nice“ participates in conjunction „and“ together with another word whose label is estimated positive

- positive, negative, neutral
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

**Example:**

- I loved the **hot** water and the **clean** bathroom.
- The **room** was **spacious** but **hot**.