RBEM: A Rule Based Approach to Polarity Detection

WISDOM’13: Workshop on Issues of Sentiment Discovery and Opinion Mining

Erik Tromp & Mykola Pechenizkiy

O2mc & Eindhoven University of Technology

The Netherlands

August 9, 2013
Motivation & Background

Sentiment analysis $\Rightarrow$ polarity detection

- Given input message $m$, learn functional mapping $p = f(m)$ where $p = \{-, +, =\}$
- Document, sentence, phrase, word level

Examples

+ I think that product A is really good
- Mr. B makes me furious!
= I am going to the mall to buy some clothes
Motivation & Background

- Polarity of adjectives, 1997
- Unsupervised polarity detection
Motivation & Background

• Sentic computing, using two-level affective common sense reasoning [E. Cambria, D. Olsher, and K. Kwok, 2012]
• Related works of Wiebe et al. use bootstrapping to label subjective patterns for subjectivity and polarity detection using high precision rules
• On social media, constructing corpora for polarity detection
• Polling for US elections using Twitter
Motivation & Background

Related work often focuses either on generic machine learning algorithms or deep learning.

Generic machine learning
- Determining what features to use is a challenge
- Might work well but often require laborious pre-processing

Deep learners
- Understanding underlying logic is cumbersome
- Maintainability is an issue
- Scalability (model training) is an issue
- Very resource and time intensive
Motivation & Background

Goal is to create a methodology that is

- Tailored towards polarity detection by design
- Transparent in use and understanding
- Competitive with other approaches
- Scalable and flexible in real life applications
- Easily adaptable and extended
- High-performance, applicable to the settings typically present in 'big data' challenges
- Portable across different application domains
Motivation & Background

Three-step approach, two-step pre-processing for polarity detection

- Language Identification
- POS Tagging
- Polarity Detection
Polarity Detection - RBEM

RBEM: Rule-Based Emission Model

Supervised algorithm using heuristic rules defined on patterns

Simple yet competitive approach, overcoming issues of deep learners, making it highly useful for real-life practical settings
Polarity Detection - RBEM

Defines 9 different pattern group classes and associates rules to them

- **Positive** Positive with no context *good, well done*
- **Negative** Negative with no context *bad, terrible*
- **Amplifier** Strengthen emission *very much, a lot*
- **Attenuator** Weaken emission *a little, a tiny bit*
- **Right Flip** Flips polarity to the right *not, no*
- **Left Flip** Flips polarity to the left *but, however*
- **Continuator** Progresses emission of polarity *and, and also*
- **Stop** Interrupt emission *punctuation*
- **Neutral** No explicit meaning, used to eliminate other patterns

[Hatzivassiloglou and McKeown, 1997]
Polarity Detection - Learning RBEM

Construct a model consisting of patterns, per-language

- Pattern consists of (word, POS-tag) pairs
- Three different types of wildcards
  - **Word wildcards** Ignore the word, use only POS-tag
  - **Single-position wildcards** Ignore the (word, POS-tag) pair but require one
  - **Multi-position wildcards** Anything holds (0 or more in between)
Polarity Detection - Learning RBEM

Construct a model consisting of patterns

- Create one model for each language separately
- Every pattern belongs to one of our pattern groups
- Supervised learning, requires manual annotation (labor intensive)

Example patterns

- **Positive** [(**good**, **ADJ**)]
- **Negative** [(**too**, **ADV**), ( _, _ )]
- **Amplifier** [(**really**, **ADV**), ( _, **ADJ**)]
Polarity Detection - Classifying with RBEM

Pre-requisites

- Language of the message must be known
- Not only words but also POS-tags must be present

Example message

This is *really not good* for me *at all*
Polarity Detection - Classifying with RBEM

Classifying consists of two steps - input is a message $m$

1. Pattern matching - Find and match patterns that occur both in $m$ and in our model
2. Rule application - Apply heuristic rules for each of the patterns found, based on pattern group
Polarity Detection - Classifying with RBEM

1. Pattern matching

1. Find all patterns in our model that occur in message $m$
2. Remove all subsumed patterns (neutral pattern important!)

**Result** A set of maximal patterns found in $m$ and their corresponding pattern groups
Polarity Detection - Classifying with RBEM

2. Rule application

Every (word, POS-tag) pair initially gets an emission value of 0

For each pattern group, alter the emission value - order is important!
2. Rule application

**Result** An emission score for each (word, POS-tag) entity $e_i$ in our message $m$

\[
\text{Polarity} = \Sigma_{i=1}^{n} em(e_i)
\]
Experiments

We want to demonstrate transparency, competitiveness, scalability, flexibility, portability, etc. of our method.

- Evaluate competitiveness of RBEM
- Demonstrate domain portability
- Evaluate real life scenarios through use cases

Construct datasets to aid our experiments
Experiments - Datasets

1. Collection method I - High-volume, noisy data
   - Use smileys for polar data, news accounts for neutral data
   - Use multiple language identifiers to filter out Dutch and English messages

2. Collection method II - Lower volume, more accurate
   - Collect English and Dutch data from Twitter using English and Dutch keywords
   - Manually annotate, using multiple annotators

For training RBEM, manually extract patterns from this training data
## Experiments - Datasets

Resulting training and test set sizes

<table>
<thead>
<tr>
<th></th>
<th>English</th>
<th>Dutch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>3614/205</td>
<td>1202/262</td>
</tr>
<tr>
<td>Negative</td>
<td>3458/200</td>
<td>1504/200</td>
</tr>
<tr>
<td>Neutral</td>
<td>4706/454</td>
<td>2099/595</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>11778/859</strong></td>
<td><strong>4805/1057</strong></td>
</tr>
</tbody>
</table>
### Experiments - Datasets

**Resulting RBEM model sizes**

<table>
<thead>
<tr>
<th>Pattern Type</th>
<th>English Count</th>
<th>Dutch Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amplifiers</td>
<td>67 (6.8%)</td>
<td>53 (7.7%)</td>
</tr>
<tr>
<td>Attenuators</td>
<td>12 (1.2%)</td>
<td>6 (0.8%)</td>
</tr>
<tr>
<td>Rightflips</td>
<td>39 (3.9%)</td>
<td>8 (1.2%)</td>
</tr>
<tr>
<td>Continuators</td>
<td>10 (1.0%)</td>
<td>4 (0.6%)</td>
</tr>
<tr>
<td>Leftflips</td>
<td>5  (0.5%)</td>
<td>2  (0.3%)</td>
</tr>
<tr>
<td><strong>Negatives</strong></td>
<td><strong>541 (54.8%)</strong></td>
<td><strong>364 (53.2%)</strong></td>
</tr>
<tr>
<td><strong>Positives</strong></td>
<td><strong>308 (31.2%)</strong></td>
<td><strong>231 (33.8%)</strong></td>
</tr>
<tr>
<td>Stops</td>
<td>0  (0.0%)</td>
<td>2  (0.3%)</td>
</tr>
<tr>
<td>Neutrals</td>
<td>6  (0.6%)</td>
<td>14 (2.0%)</td>
</tr>
</tbody>
</table>
Experiments - RBEM Accuracy

NB  Naive-Bayes
SVM Support Vector Machines ($SVM_{light}$)
PPC Prior-Polarity Classifier using SentiWordNet as lexicon
AB AdaBoost using decision stumps

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB - All Tokens</td>
<td>0.616</td>
</tr>
<tr>
<td>SVM - All Patterns</td>
<td>0.637</td>
</tr>
<tr>
<td>RBEM</td>
<td>0.724</td>
</tr>
<tr>
<td>PPC - Using SentiWordNet</td>
<td>0.681</td>
</tr>
<tr>
<td>AB - POS-tags - 50 stumps</td>
<td>0.723</td>
</tr>
</tbody>
</table>
Experiments - Domain Portability

- Apply generically trained model on domain-specific data
  - Domain: Television broadcasts
- Consult domain experts for verification
- Extract common errors
- Refine model by inserting or removing patterns
Experiments - Domain Portability

Example corrections

1. Different shows broadcasted on television during prime-time
   - The verb *to cry* is generally negative, so is *a pity*
   - Crying with respect to a television show is a good thing since it triggers emotion
   - The pattern *a pity* was often used to indicate it was a pity the show was over

2. Dutch television show *Goede tijden, slechte tijden* ('good times, bad times')
   - *Good* is a positive pattern, *bad* a negative pattern
   - Mark *good times, bad times* as neutral
Experiments - Domain Portability

Results, evaluated by same domain experts

1. Number of errors reduced by 64%
2. Number of errors reduced by 87.5%
Experiments - Use Cases

Demonstrate the usefulness of RBEM, in operation within a framework applied at real life scenarios

1. SentiCorr - Application of sentiment analysis on personal correspondence. Should be flexible and tailored towards a specific individual
   • As an aid for determining stress factors
2. Mobile SentiCorr - Portability of application to mobile analysis
   • Resource limitation, hence scalability and flexibility are important
3. Emotion Tracker (In Dutch: Emotiepeiler, see http://www.emotiepeiler.nl) - A showcase website of generic, real-time application of our sentiment analysis platform. On average 50 tweets/second per machine ⇒ scalability important
Conclusions

• Main contribution is RBEM, polarity detection that is
  • Transparent, competitive, scalable, extendable, flexible
  • Applicable in broad real life applications

• Constructed datasets for evaluation, released to public (For now, see http://www.win.tue.nl/~mpechen/)

• Evaluated RBEM accuracy and domain portability
Future Work

- Automated extraction and evaluation of patterns ⇒ semi-supervised RBEM
- Currently working on emotion classification, extending beyond polarity
- Investigate (relevance) feedback mechanisms
- Investigate applicability to non-latin written languages
  - Currently looking at Arabic
  - Also look at Chinese, Japanese, Russian, etc.
- CINLP: RBEM integration into ubiquitous sentiment analysis platform
- Monitoring mechanisms (eg. for concept drift, anomaly detection, etc.)

- PhD: Authorship profiling
Discussion

Feel free to contact us!

Erik Tromp
Data Scientist - Open Source Intelligence Adversitement O2mc
erik.tromp@o2mc.eu
linkedin.com/in/eriktromp
twitter.com/Eriku_T

Mykola Pechenizkiy
Assistant Professor - Web Engineering Eindhoven University of Technology
m.pechenizkiy@tue.nl
linkedin.com/in/mpechen
win.tue.nl/~mpechen