Feature Specific Sentiment Analysis of Reviews

Subhabrata Mukherjee and Pushpak Bhattacharyya
Dept. of Computer Science and Engineering,
IIT Bombay

13th International Conference on Intelligent Text Processing and Computational Intelligence - CICLING 2012,
New Delhi, India, March, 2012
Sentiment Analysis is *always* with respect to a particular entity or feature

Feature may be *implicit* or *explicit*

This work concerns *explicit feature*
I have an ipod and it is a great buy but I'm probably the only person that dislikes the iTunes software.

Here the sentiment w.r.t ipod is positive whereas that respect to software is negative.
An entity may be analyzed from the point of view of multiple features

- Entity – Titanic
- Features – Music, Direction, Plot etc.

Given a sentence how to identify the set of features?
- Each sentence can contain **multiple** features and **mixed** opinions (positive and negative)

- Reviews mixed from various domains

- No prior information about set of features except the **target feature**
MAIN FEATURES OF THE ALGORITHM

- Does not require any prior information about any domain
- Unsupervised – But need a small untagged dataset to tune parameters
- Does not require any prior feature set
- Groups set of features into separate clusters which need to be pruned or labeled
“More closely related words come together to express an opinion about a feature”
“I want to use Samsung which is a great product but am not so sure about using Nokia”.

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitivity related.

Here “great” and “product” are more related to Samsung than they are to Nokia.

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
“I want to use Samsung which is a great product but am not so sure about using Nokia”.

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitively related.

Here “great” and “product” are more related to Samsung than they are to Nokia

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
Hypothesis Example

“I want to use Samsung which is a **great** product but am not so sure about using Nokia**.”

- Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transительно related.
- **Here “great” and “product” are more related to Samsung than they are to Nokia**
- Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
“I want to use Samsung which is a great product but am not so sure about using Nokia.”

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitively related.

Here “great” and “product” are more related to Samsung than they are to Nokia.

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
“I want to use Samsung which is a great product, but am not so sure about using Nokia.”

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitively related.

Here “great” and “product” are more related to Samsung than they are to Nokia

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
“I want to use Samsung which is a great product but am not so sure about using Nokia.”

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitively related.

Here “great” and “product” are more related to Samsung than they are to Nokia

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
“I want to use Samsung which is a great product but am not so sure about using Nokia.”

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitively related.

Here “great” and “product” are more related to Samsung than they are to Nokia.

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
“I want to use Samsung which is a great product but am not so sure about using Nokia”.

Here “great” and “product” are related by an adjective modifier relation, “product” and “Samsung” are related by a relative clause modifier relation. Thus “great” and “Samsung” are transitively related.

Here “great” and “product” are more related to Samsung than they are to Nokia.

Hence “great” and “product” come together to express an opinion about the entity “Samsung” than about the entity “Nokia”
I have an **ipod** and it is a great buy but I'm probably the only person that dislikes the **iTunes software**.
Example of a Review

- I have an **ipod** and it is a great buy but I'm probably the only person that dislikes the **iTunes software**.
Example of a Review

- I have an **ipod** and it is a great buy but I'm probably the only person that dislikes the **iTunes** software.
Example of a Review

- I have an **ipod** and it is a **great** buy but I'm probably the only person that dislikes the **iTunes** **software**.
Example of a Review

- I have an **ipod** and it is a great buy but I'm probably the only person that dislikes the **iTunes** software.
Feature Extraction: Domain Info
Not Available
Initially, all the Nouns are treated as features and added to the feature list $F$. 
Initially, all the Nouns are treated as features and added to the feature list $F$.

$F = \{ \text{ipod, buy, person, software} \}$
Initially, all the Nouns are treated as features and added to the feature list $F$.

$F = \{ \text{ipod, buy, person, software} \}$

Pruning the feature set
- Merge 2 features if they are strongly related
Initially, all the Nouns are treated as features and added to the feature list $F$.

$F = \{ ipod, buy, person, software \}$

Pruning the feature set
- Merge 2 features if they are strongly related

“buy” merged with “ipod”, when target feature = “ipod”,
- “person, software” will be ignored.
Feature Extraction : Domain Info
Not Available

- Initially, all the Nouns are treated as features and added to the feature list $F$.

- $F = \{ \text{ipod, buy, person, software} \}$

- Pruning the feature set
  - Merge 2 features if they are strongly related

- “buy” merged with “ipod”, when target feature = “ipod”,
  - “person, software” will be ignored.

- “person” merged with “software”, when target feature = “software”
  - “ipod, buy” will be ignored.
Relations

- **Direct Neighbor Relation**
  - Capture **short range dependencies**
  - Any 2 consecutive words (such that none of them is a StopWord) are directly related
  - Consider a sentence $S$ and 2 consecutive words $w_i, w_{i+1}$.
  - If $w_i, w_{i+1} \notin \text{Stopwords}$, then they are directly related. $w_i, w_{i+1} \in S$

- **Dependency Relation**
  - Capture **long range dependencies**
  - Let $Dependency\_Relation$ be the list of **significant relations**.
  - Any 2 words $w_i$ and $w_j$ in $S$ are directly related, if
    $\exists D_i \text{ s.t. } D_i(w_i, w_j) \in Dependency\_Relation$
Given a sentence $S$, let $W$ be the set of all words in the sentence $S$.

A Graph $G(W,E)$ is constructed such that any $w_i, w_j \in W$ are directly connected by $e_k \in E$, if \[ \exists R_l \text{ s.t. } R_l(w_i, w_j) \in R. \]
i. Initialize $n$ clusters $C_i \ \forall i = 1..n$

ii. Make each $f_i \in F$ the clusterhead of $C_i$. The target feature $f_t$ is the clusterhead of $C_t$. Initially, each cluster consists only of the clusterhead.
iii. Assign each word $w_j \in S$ to cluster $C_k$

\[ s.t. \quad k = \arg \min_{i \in n} \text{dist}(w_j, f_i), \]

Where $\text{dist}(w_j, f_i)$ gives the number of edges, in the shortest path, connecting $w_j$ and $f_i$ in $G$. 
iv. Merge any cluster $C_i$ with $C_t$ if, 
\[ \text{dist}(f_i, f_t) < \theta, \]
Where $\theta$ is some threshold distance.

v. Finally the set of words $w_i \in C_t$ gives the opinion expression regarding the target feature $f_t$. 
Clustering
Clustering
Clustering
Clustering
Clustering
Clustering
Clustering

7/23/2013
Clustering
Clustering
Evaluation – Dataset 1

- 2500 sentences

- Varied domains like antivirus, camera, dvd, ipod, music player, router, mobile

- Each sentence tagged with a feature and polarity w.r.t the feature

- Acid Test
  - Each Review has a mix of positive and negative comments
Parameter Learning

- Dependency Parsing uses approx. 40 relations.

- Relation Space – \((2^{40} - 1)\)

- Infeasible to probe the entire relation space.

- Fix relations certain to be significant
  - nsubj, nsubjpass, dobj, amod, advmod, nn, neg

- Reject relations certain to be non-significant
This leaves around 21 relations some of which may not be significant.

Compute Leave-One-Relation out accuracy over a training set.

Find the relations for which there is significant accuracy change.
# Ablation test

<table>
<thead>
<tr>
<th>Relations</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>63.5</td>
</tr>
<tr>
<td>Dep</td>
<td>67.3</td>
</tr>
<tr>
<td>Rcmd</td>
<td>65.4</td>
</tr>
<tr>
<td>xcomp, conj_and ccomp, iobj</td>
<td>61.5</td>
</tr>
<tr>
<td>advcl, appos, csubj, abbrev, infmod, npavmod, rel, acomp, agent, csubjpass, partmod, pobj, purpcl, xsubj</td>
<td>63.5</td>
</tr>
</tbody>
</table>
## Ablation test

<table>
<thead>
<tr>
<th>Relations</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>63.5</td>
</tr>
<tr>
<td>Dep</td>
<td>67.3</td>
</tr>
<tr>
<td>Rcmd</td>
<td>65.4</td>
</tr>
<tr>
<td>xcomp, conj_and ccomp, iobj</td>
<td>61.5</td>
</tr>
<tr>
<td>advcl, appos, csubj, abbrev, infmod, npavmod, rel, acomp, agent, csubjpass, partmod, pobj, purpcl, xsubj</td>
<td>63.5</td>
</tr>
</tbody>
</table>
### Ablation test

<table>
<thead>
<tr>
<th>Relations</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>63.5</td>
</tr>
<tr>
<td>Dep</td>
<td>67.3</td>
</tr>
<tr>
<td>Rcmod</td>
<td>65.4</td>
</tr>
<tr>
<td>xcomp, conj_and</td>
<td>61.5</td>
</tr>
<tr>
<td>ccomp, iobj</td>
<td></td>
</tr>
<tr>
<td>advcl, appos, csubj,</td>
<td>63.5</td>
</tr>
<tr>
<td>abbrev, infmod, npavmod, rel, acomp, agent, csubjpash, partmod, pobj, purpcl, xsubj</td>
<td></td>
</tr>
</tbody>
</table>
Leaving out *dep* improves accuracy most.
<table>
<thead>
<tr>
<th>Relation Set</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>With Dep+Rcmod</td>
<td>66</td>
</tr>
<tr>
<td>Without Dep</td>
<td>69</td>
</tr>
<tr>
<td>Without Rcmod</td>
<td>67</td>
</tr>
<tr>
<td>Without Dep+Rcmod</td>
<td>68</td>
</tr>
</tbody>
</table>

- Leaving out *dep* improves accuracy most
Inter cluster distance

<table>
<thead>
<tr>
<th></th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>67.85</td>
</tr>
<tr>
<td>3</td>
<td><strong>69.28</strong></td>
</tr>
<tr>
<td>4</td>
<td>68.21</td>
</tr>
<tr>
<td>5</td>
<td>67.4</td>
</tr>
</tbody>
</table>
## Inter cluster distance

<table>
<thead>
<tr>
<th></th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>67.85</td>
</tr>
<tr>
<td><strong>3</strong></td>
<td><strong>69.28</strong></td>
</tr>
<tr>
<td>4</td>
<td>68.21</td>
</tr>
<tr>
<td>5</td>
<td>67.4</td>
</tr>
</tbody>
</table>
## Lexicon based classification

<table>
<thead>
<tr>
<th>Domain</th>
<th>Baseline 1 (%)</th>
<th>Baseline 2 (%)</th>
<th>Proposed System (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antivirus</td>
<td>50</td>
<td>56.82</td>
<td>63.63</td>
</tr>
<tr>
<td>Camera 1</td>
<td>50</td>
<td>61.67</td>
<td>78.33</td>
</tr>
<tr>
<td>Camera 2</td>
<td>50</td>
<td>61.76</td>
<td>70.58</td>
</tr>
<tr>
<td>Camera 3</td>
<td>51.67</td>
<td>53.33</td>
<td>60.00</td>
</tr>
<tr>
<td>Camera 4 (Nikon)</td>
<td>52.38</td>
<td>57.14</td>
<td>78.57</td>
</tr>
<tr>
<td>DVD</td>
<td>52.21</td>
<td>63.23</td>
<td>66.18</td>
</tr>
<tr>
<td>IPOD</td>
<td>50</td>
<td>57.69</td>
<td>67.30</td>
</tr>
<tr>
<td>Mobile 1</td>
<td>51.16</td>
<td>61.63</td>
<td>66.28</td>
</tr>
<tr>
<td>Mobile 2</td>
<td>50.81</td>
<td>65.32</td>
<td>70.96</td>
</tr>
<tr>
<td>Music Player 1</td>
<td>50.30</td>
<td>57.62</td>
<td>64.37</td>
</tr>
<tr>
<td>Music Player 2</td>
<td>50</td>
<td>60.60</td>
<td>67.02</td>
</tr>
<tr>
<td>Router 1</td>
<td>50</td>
<td>58.33</td>
<td>61.67</td>
</tr>
<tr>
<td>Router 2</td>
<td>50</td>
<td>59.72</td>
<td>70.83</td>
</tr>
</tbody>
</table>
Lexicon based classification

<table>
<thead>
<tr>
<th>Domain</th>
<th>Baseline 1 (%)</th>
<th>Baseline 2 (%)</th>
<th>Proposed System (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antivirus</td>
<td>50</td>
<td>56.82</td>
<td>63.63</td>
</tr>
<tr>
<td>Camera 1</td>
<td>50</td>
<td>61.67</td>
<td>78.33</td>
</tr>
<tr>
<td>Camera 2</td>
<td>50</td>
<td>61.76</td>
<td>70.58</td>
</tr>
<tr>
<td>Camera 3</td>
<td>51.67</td>
<td>53.33</td>
<td>60.00</td>
</tr>
<tr>
<td>Camera 4(Nikon)</td>
<td>52.38</td>
<td>57.14</td>
<td>78.57</td>
</tr>
<tr>
<td>DVD</td>
<td>52.21</td>
<td>63.23</td>
<td>66.18</td>
</tr>
<tr>
<td>IPOD</td>
<td>50</td>
<td>57.69</td>
<td>67.30</td>
</tr>
<tr>
<td>Mobile 1</td>
<td>51.16</td>
<td>61.63</td>
<td>66.28</td>
</tr>
<tr>
<td>Mobile 2</td>
<td>50.81</td>
<td>65.32</td>
<td>70.96</td>
</tr>
<tr>
<td>Music Player 1</td>
<td>50.30</td>
<td>57.62</td>
<td>64.37</td>
</tr>
<tr>
<td>Music Player 2</td>
<td>50</td>
<td>60.60</td>
<td>67.02</td>
</tr>
<tr>
<td>Router 1</td>
<td>50</td>
<td>58.33</td>
<td>61.67</td>
</tr>
<tr>
<td>Router 2</td>
<td>50</td>
<td>59.72</td>
<td>70.83</td>
</tr>
</tbody>
</table>
# Overall accuracy

<table>
<thead>
<tr>
<th>Method</th>
<th>Average Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 1</td>
<td>50.35</td>
</tr>
<tr>
<td>Baseline 2</td>
<td>58.93</td>
</tr>
<tr>
<td>Proposed System</td>
<td>70.00</td>
</tr>
</tbody>
</table>
## Overall accuracy

<table>
<thead>
<tr>
<th>Method</th>
<th>Average Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 1</td>
<td>50.35</td>
</tr>
<tr>
<td>Baseline 2</td>
<td>58.93</td>
</tr>
<tr>
<td>Proposed System</td>
<td>70.00</td>
</tr>
</tbody>
</table>
Evaluation – Dataset 2

- Extracted 500 sentences
- Varied domains like camera, laptop, mobile
- Each sentence tagged with a feature and polarity w.r.t the feature
### Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 1</td>
<td>68.75</td>
</tr>
<tr>
<td>Baseline 2</td>
<td>61.10</td>
</tr>
<tr>
<td>CFACTS-R</td>
<td>80.54</td>
</tr>
<tr>
<td>CFACTS</td>
<td>81.28</td>
</tr>
<tr>
<td>FACTS-R</td>
<td>72.25</td>
</tr>
<tr>
<td>FACTS</td>
<td>75.72</td>
</tr>
<tr>
<td>JST</td>
<td>76.18</td>
</tr>
<tr>
<td>Proposed System</td>
<td>80.98</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 1</td>
<td>68.75</td>
</tr>
<tr>
<td>Baseline 2</td>
<td>61.10</td>
</tr>
<tr>
<td>CFACTS-R</td>
<td>80.54</td>
</tr>
<tr>
<td><strong>CFACTS</strong></td>
<td><strong>81.28</strong></td>
</tr>
<tr>
<td>FACTS-R</td>
<td>72.25</td>
</tr>
<tr>
<td>FACTS</td>
<td>75.72</td>
</tr>
<tr>
<td>JST</td>
<td>76.18</td>
</tr>
<tr>
<td><strong>Proposed System</strong></td>
<td><strong>80.98</strong></td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 1</td>
<td>68.75</td>
</tr>
<tr>
<td>Baseline 2</td>
<td>61.10</td>
</tr>
<tr>
<td>CFACTS-R</td>
<td>80.54</td>
</tr>
<tr>
<td><strong>CFACTS</strong></td>
<td><strong>81.28</strong></td>
</tr>
<tr>
<td>FACTS-R</td>
<td>72.25</td>
</tr>
<tr>
<td>FACTS</td>
<td>75.72</td>
</tr>
<tr>
<td>JST</td>
<td>76.18</td>
</tr>
<tr>
<td><strong>Proposed System</strong></td>
<td><strong>80.98</strong></td>
</tr>
</tbody>
</table>
CONCLUSIONS

- Incorporating feature specificity improves sentiment accuracy.

- Dependency Relations capture long range dependencies as is evident from accuracy improvement.

- Work to be extended for *implicit features and domain dependent sentiment*. 