1. Problem Statement
Automatic cyberbullying detection methods are unfit for real-world applications [3]. This is largely due to:
• Unreliable data: inconsistent criteria, [2,3,4] context-blind annotations, [3] class imbalance [2,3]
• Coarse features: bag-of-words (BoW) methods lack nuance and cannot adapt to language change

Goal 1: Produce a reliable dataset of labeled cyberbullying cases within Twitter threads.
Goal 2: Train a cyberbullying classifier from a refined set of social features.

2. Data Collection
• Scrape: 1.3 million tweets from Stream API
• Filter: English, @ mentions, non RTs, visible threads, hate speech / offensive language [1]
• 6,897 message threads
• Collect user data: account information (friends, following) and 6 months of each timeline.

3. Annotation Task
MTurk study: 3 annotations per message thread
• Label author & target @handles for each tweet
Given the full message thread and up to 15 recent mentions, provide labels for 5 criteria
1) Aggressive language: confrontational, derogatory, insulting, threatening, hostile, violent, hateful, or sexually abusive language directed towards individual or group [2,3,6]
2) Repetition: 2+ aggressive messages [2,3,4]
3) Harmful intent: author intends to tear down or disadvantage the target user [3,4,5]
4) Visibility among peers: one other user has liked, quoted, retweeted or responded to the author [3]
5) Power Imbalance: does the author or target have greater social advantage / perceived authority? [2,4]

3.1 Annotation Task

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Class Balance</th>
<th>Inter-annotator Agreement</th>
<th>Cyberbullying Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>aggression</td>
<td>0.23</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>repetition</td>
<td>0.18</td>
<td>0.27</td>
<td></td>
</tr>
<tr>
<td>harmful intent</td>
<td>0.42</td>
<td>0.22</td>
<td></td>
</tr>
<tr>
<td>visibility among peers</td>
<td>0.51</td>
<td>0.07</td>
<td></td>
</tr>
<tr>
<td>target power</td>
<td>0.37</td>
<td>0.11</td>
<td></td>
</tr>
<tr>
<td>author power</td>
<td>0.10</td>
<td>-0.02</td>
<td></td>
</tr>
<tr>
<td>equal power</td>
<td>0.22</td>
<td>-0.09</td>
<td></td>
</tr>
<tr>
<td>cyberbullying</td>
<td>0.18</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

Advantages: clear criteria, flexible cyberbullying definition, context-aware annotations, more balanced class distributions.

4. Feature Engineering
Baseline Features
• Text: N-Grams, LIWC, VADER, Fleisch-Kincaid Reading Ease [1,3,5]
• User: Friend/following counts, verified status, number of posts

Thread Features
• Visibility
  • Message count, reply message count, reply user count, max author favorites, max author RTs
• Aggression
  • Aggressive message count, aggressive author message count, aggressive user count [1]

Timeline Features
• Message Behavior
  • Directed message counts
• Mentions overlap (Jaccard)
• Language Models
  • New-words ratio
• Cross-entropy
  • $H(m) = \frac{1}{\log base} \sum Log P(b_i)$
  for message m with bigrams $b_i$, $b_j \neq b_i$
• Timeline similarity
  • $\cos \theta = \frac{\vec{A} \cdot \vec{T}}{\| \vec{A} \| \cdot \| T \|}$
  for author timeline $A$ and target timeline $T$

Network Features
• Neighborhood Overlap
  • $|C(A) \cap C(T)| / |C(A) \cup C(T)|$
  for author $a$ and target $t$, $R(A)$ is the neighborhood set of user $a$

5. Model Evaluation

6. Conclusions

• Text-based methods can reliably detect aggressive language
• Social features are better suited for detecting repetition, visibility among peers, and power imbalance
• Classifiers are not yet ready for the real world [3,4]
• Future Work: increase performance, build new features, detect social roles, measure efficiency (run time, number of API calls, etc.)

References

If interested contact Caleb Ziemz: cziems@emory.edu
Project website: reu.usc.edu
Work performed under REU Site program supported by NSF grant #1659886.