Identifying Secondary Crashes for Traffic Incident Management (TIM) Programs

Xu Zhang
Kentucky Transportation Center
9/14/2018
TIM Team

Eric Green  Mei Chen  Reg Souleyrette
Secondary Crash Example
Outline

• Kentucky TIM Overview
• Secondary Crash Identification Methods
• Adapting Text Mining Techniques
• Results and Discussions

*Paper under review by TRB*
Kentucky TIM Overview

• Part of FHWA’s EDC 4 Initiative
• Formed Incident Management Task Force with representatives from FHWA, KYTC, KSP, local agencies and KTC
• Focusing on identifying resources needed to compute three major measures
  • Roadway clearance time
  • Incident clearance time
  • Secondary crashes
KSP Secondary Crash Reporting

- “Secondary Collision” field was added in 2007
- Additional help button was added in 2013
- Miscoded secondary crashes (type I errors or false positives)
- Non-reported secondary crashes (type II errors or false negatives)
Secondary Crash Identification Methods

- Spatiotemporal approach with fixed thresholds
- Traffic theory models
- Crash impact zone using speeds from loop detectors or probe data

Analyzing Crash Narratives

```
Traffic was moving very slow due to an injury accident just ahead. The slow lane was the only lane open for traffic. Unit 1 made contact with Unit 2 as Unit 1 was starting to change lanes. Unit 1 was unable to see Unit 2 stopped to his right.
```

“Traffic was moving **very slow due to an injury accident just ahead**. The slow lane was the only lane open for traffic. Unit 1 made contact with Unit 2 as Unit 1 was starting to change lanes. Unit 1 was unable to see Unit 2 stopped to his right.”
Secondary Crash Identification

- Spatiotemporal approach to identify candidate primary-secondary crash pairs
- OCR tool to extract crash narratives into text
- Keywords search to exclude unlikely secondary crashes
- Manually review the remaining crash narratives
- Confirmed 1605 secondary crashes out of 5154 crashes reviewed for 2015-2017

Can we let machines do this?
Text Mining Applications

• Spell checking in Word
• Junk email filtering
• Google search
• Fake news detection
• Question answering, e.g. Siri, Google Assistant, Alexa
• Language translation
Text Mining Process

• Tokenization (e.g. bag of words)
• Counting
• Vectorization
• Normalization
  • TF-IDF
Secondary Crash Classifier

- Logistic Regression
- Random Forest
- Naïve Bayes
- Support Vector Machines
Performance Metrics

**Accuracy** = \( \frac{\text{True Negatives} + \text{True Positives}}{\text{Total Number of Crashes}} \) \times 100

**Precision** = \( \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \) \times 100

**Recall** = \( \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \) \times 100

\( F_1 = \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \) \times 2
## Model Performance

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>84</td>
<td>71.3</td>
<td>78.4</td>
<td><strong>74.7</strong></td>
</tr>
<tr>
<td>Random Forest</td>
<td>82.3</td>
<td><strong>72.1</strong></td>
<td>67.4</td>
<td>69.7</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>76.5</td>
<td>59.4</td>
<td>69.4</td>
<td>64</td>
</tr>
<tr>
<td>Support Vector Machine</td>
<td>76.3</td>
<td>57.3</td>
<td><strong>83.9</strong></td>
<td>68.1</td>
</tr>
</tbody>
</table>
## Effect of Tokenization

<table>
<thead>
<tr>
<th>N-grams</th>
<th>Number of Features</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9302</td>
<td>84</td>
<td>71.3</td>
<td>78.4</td>
<td>74.7</td>
</tr>
<tr>
<td>2</td>
<td>81345</td>
<td>85.5</td>
<td>76.1</td>
<td>75.8</td>
<td>75.9</td>
</tr>
<tr>
<td>3</td>
<td>190088</td>
<td>81.7</td>
<td>73.7</td>
<td>60.6</td>
<td>66.5</td>
</tr>
<tr>
<td>(1, 2)</td>
<td>90647</td>
<td>86.1</td>
<td>76.2</td>
<td>78.4</td>
<td>77.3</td>
</tr>
<tr>
<td>(2, 3)</td>
<td>271433</td>
<td>85.4</td>
<td>77.1</td>
<td>72.9</td>
<td>75</td>
</tr>
<tr>
<td>(1, 3)</td>
<td>280735</td>
<td>85.7</td>
<td>75.2</td>
<td>78.4</td>
<td>76.8</td>
</tr>
</tbody>
</table>

---

===> Uni-gram based Logistic Regression
Importance of Words

due 6.6
previous 5.2
another 5.0
separate 4.4
earlier 4.3
because 3.5
unrelated 3.2
further 3.2
ahead 3.1
already 3.0
blocking 2.8
distracted 2.8
just 2.8
looking 2.8
blocked 2.7
involved 2.7
working 2.7
emergency 2.7
“Both units were stopped in traffic due to another unrelated accident. Unit one rolled slowly backwards and struck unit two”
False Positive

“Prior to the collision Unit 1 was north on I-XX and Unit 2 was stopped in traffic on I-XX in front of Unit 1. Unit 1 operator stated that she was distracted by another vehicle who was following her too closely when she struck the rear passenger side of Unit 2. Unit 2 stated that she was stopped in traffic when Unit 1 struck the rear passenger side of Unit 2. Unit 1 could not provide proof of insurance and was cited to court.”
False Positive

“Unit 1 was involved in a single vehicle collision on I-XX around the 135MM westbound. The collision occurred due to inclimate weather and slick road surface from snowfall.”
False Negative

“Unit 1 was heading N/B on XX Road attempting to make a u-turn and head S/B on XX Road at YY Circle. Unit2 was headed S/B on XX Road when unit1 pulled into her direction of travel failing to yield the right of way. Unit1 stated that he never saw Unit2. There was heavier than usual traffic congestion from an accident at US-ZZ and XX Road and traffic was being rerouted.”
Conclusions

• Text mining is promising for secondary crash identification
• Scalable to longer time periods and larger geographical areas
• Transferrable to other states
• Continuous effort to include more training data and fine-tune the model
Questions?

Email: xuzhang_uk@uky.edu