Forum post classification to support forensic investigations of illegal trade on the Dark Web

System & Network Engineering (MSc)
Research Project 2

Supervisors: Martijn Spitters Stefan Verbruggen
Diana Rusu diana.rusu@os3.nl
Motivation

- Illegal business thriving under the DeepWeb
- Processing large amounts of data needs (semi) automatization
- Keyword matching is not sufficient for classification tasks
- Current techniques require large training sets
Research Question

In the context of grouping DarkWeb marketplaces forum posts into relevant categories useful for forensic investigators

Can we boost the classification process using semantic word representations, in order to reduce the required amount of training samples?

Subquestions:

1. What methods can be inferred to exploit the word representations for classifying sparse, short forum posts on discussion forums, using few examples?
2. What is the accuracy of the proposed methods and how can it be improved?
Word2Vec

- It represents a word as a vector (word representations)
- Given a word it will predict n similar words around it
- Given some words it will give the appropriate word in that context (animated image [1])
- Creates a “semantic space” from large amount of data
- Based on
  - skip-gram
  - CBOW (continuous bag-of-words)
Experimental Data

Dataset provided by TNO, aggregated from different forums that accompany DeepWeb marketplaces such as Agora or Evolution:

<table>
<thead>
<tr>
<th></th>
<th>Raw Posts</th>
<th>Tokenized Posts (after preprocessing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Posts</td>
<td>1954508</td>
<td>1447029</td>
</tr>
<tr>
<td>Words</td>
<td>138310824</td>
<td>42835813</td>
</tr>
</tbody>
</table>
Taxonomy Class
Approach

- DeepWeb dataset
- Select & Build Training Set
- Create Word2Vec Model
- Retrieve the n most similar posts/(query|corpus)
- Create Vector/each post
- Create Average vector/Class

Start Point
Intermediate Point
Experiments - Setup

- Training Set
- Cosine Similarity
- Expand Training Set
- SVM
- Test Set
- Test Set 1 = 60 random posts
- DeepWeb dataset
- Test Set 2 = 100 random post

Intermediate Point

End Point
Experiments

All of the following results provided were accomplished by having a single-class assigned for each post.
Experiments - Example 1

Human label - "hard_drugs"

Post 97 1072694 fakename wrote: i dont like street deals so i buy only here and another markets but need a fair deal. I gave you a vendor, whose prices are decent for an online market. And there are a shittonne of vendors online selling the Nijntje pills ... themostseekrit contact details upon request But I see nothing, no eyes ... no eyes on me.

-------------------------------------
***********Highest Rank(bottom-up)**************
TOP 36: greetings - 0.22749844193458557,

TOP5: trading_scamming - 0.8590390682220459,
TOP7: vendors - 0.8627676367759705,
TOP6: trading_shipping - 0.8668627142906189
TOP5: financial_carding - 0.8688409924507141,

TOP4: hard_drugs - 0.8711443543434143,
TOP3: other - 0.8717963695526123
TOP 2: trading_feedback - 0.8815533518791199,

TOP 1: trading_recommendation - 0.8951979279518127

-- The example above uses Cosine Similarity when testing with 100 Test Set Sample--
Results

**Accuracy**: percentage of test instances for which the correct label was ranked as #1 in cosine similarity or SVM learning method

Y-axis: Accuracy in %
Results

Excluding “other” class label:

Y-axis: Accuracy in %
Results

When expanding the training set (applied in case of Cosine Similarity):

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy (in %)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Test Set 1 - 60</td>
</tr>
<tr>
<td></td>
<td>Random Posts</td>
</tr>
<tr>
<td>Cosine Similarity</td>
<td>14.0350877193</td>
</tr>
</tbody>
</table>
Results

Plot 1: The accuracy of the Cosine Similarity between the AverageVector Class and the Vector Test class increases significantly if searching in TOP_4 the “human” labeled class.

Y-axis - Accuracy in %  
X-axis - TOP classes
Plot 2: The accuracy of the Cosine Similarity between the same samples, where it can be seen an accuracy of TOP 4 at ~50%, while in the case of extending the initial training set ~40%.
Conclusions

- **Cosine Similarity**, using word representations, provides ~20% accuracy from the first run (TOP1) based on the experiments conducted (single-class label for each post), while **SVM** shows a better result with ~39% accuracy.

- **Cosine Similarity** improves significantly its accuracy if searching in TOP4 values assigned by the classifier, the “human” labeled class. In this case will achieve ~50% accuracy. SVM needs to be tested for the TOPn classes (report).

- In practice, based on the results, if improving a small training set with the correct multi-class labeling for each post it is feasible to use word representations as futures for a classifier, in order to get a quick thematic insight over the discussion forums which reside under the Dark Web.
Future Work

- Training Set has to be reviewed by at least 2 persons
- Expand the Taxonomy class
- Integrate this classifier into the DarkWebMonitor portal (darkwebmonitor.eu)
Questions?
References