Extracting emerging knowledge from social media

Jae Hee Lee (COMP3740)
(Supervisor: Dongwoo Kim)
18 May 2018
Motivation

• Aiming to construct multimedia knowledge graph, a part of ‘Picturing Knowledge’ project.
• Enhancing the existing knowledge graph developed at the ANU Computational Media Lab
• Focusing on extracting relations from multimedia source
Background: Knowledge Graph

Representing knowledge in a graph to connect related information
Background: Knowledge Base

Database for knowledge sharing
"Barack Obama" query to DBpedia

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>dbo:birthPlace</td>
<td>dbr:Hawaii, dbr:Honolulu, dbr:Kapiolani_Medical_Center_for_Women_and_Children</td>
</tr>
<tr>
<td>dbo:orderInOffice</td>
<td>44th President of the United States</td>
</tr>
<tr>
<td>dbo:party</td>
<td>dbr:Democratic_Party_(United_States)</td>
</tr>
<tr>
<td>dbo:region</td>
<td>dbr:Illinois</td>
</tr>
<tr>
<td>dbo:religion</td>
<td>dbr:Protestantism</td>
</tr>
<tr>
<td>dbo:residence</td>
<td>dbr:White_House</td>
</tr>
<tr>
<td>dbo:seniority</td>
<td>United States Senator</td>
</tr>
<tr>
<td>dbo:soundRecording</td>
<td>dbr:Barack_Obama__1, dbr:Barack_Obama__2</td>
</tr>
<tr>
<td>dbo:spouse</td>
<td>dbr:Michelle_Obama</td>
</tr>
<tr>
<td>dbo:successor</td>
<td>dbr:Kwame_Raoul, dbr:Roland_Burris</td>
</tr>
<tr>
<td>dbo:thumbnail</td>
<td>wiki-commons:Special:FilePath/President_Barack_Obama.jpg?width=300</td>
</tr>
<tr>
<td>dbo:vicePresident</td>
<td>dbr:Joe_Biden</td>
</tr>
</tbody>
</table>
Background: OpenIE

Extraction of relation tuples from text by leveraging the sentence structure.

Born in a small town, she took the midnight train going anywhere

Each clause then maximally shortened

She took the midnight train

She took midnight train

Fragments are segmented into OpenIE triples

(She;took;midnight train)

Split each sentence into a set of entailed clauses

she took the midnight train going anywhere

Born in a small town, she took the midnight train

Born in a town, she took the midnight train
Background: Neural Relation Extraction (NRE)

Serves the same purpose as OpenIE but uses neural network to extract relations. Two neural network models are implemented:

• recurrent neural network (RNN); and
• convolutional neural network (CNN).
Background: Neural Network
Convolutional Neural Network (CNN)

• Often used for classifying input images into categories.
• Four main layers:
  – Convolution layer
  – Non-linearity function layer
  – Max-pooling layer
  – Fully connected layer
CNN

Consider each image as a matrix of pixel values.
CNN: Convolution Layer

- Extracts features from an image.
- Green matrix: 5 x 5 image pixels
- Yellow matrix: 3 x 3 matrix (a filter)
- From the top-left, CNN slides the filter by 1 pixel (a stride) and perform some calculation (e.g. element-wise multiplication).
- The final output is called ‘feature map’
CNN: Non-linearity Layer

- If negative value, it returns 0 and positive values stay unchanged.
- The feature map becomes the rectified feature map.

Rectified Linear Unit (ReLU)

Output = Max(zero, Input)
CNN: Max-pooling layer

- This layer further reduces the dimensionality of each feature map but retains most important information.
- The max-pooling gets the maximum value. Like the convolution layer, we slide over the feature map.
CNN: Fully Connected Layer

• The output of max-pooling layer represents features of the input image.
• The purpose is to use these features for classifying the input into various classes.
• Therefore, this layer represents a feature vector for the input.
• The sum of output probabilities from the layer is 1 by applying Softmax function that generates a probability distribution over N different outcomes.
Work Done

• Data preparation for NRE (extracted from ElasticSearch DB)
• OpenIE implementation
• NRE implementation
• Graph Visualization
OpenIE + Visualization
Implementation details (OpenIE)

• Used Java with Stanford NLP tools
• FrameNet and DBpedia were used to extract the canonical form of entities-relation tuples.
  – Semafor was used as an API to use FrameNet data.
Implementation details (Graph)

- Python used for backend (Django framework)
- JavaScript with D3.js library for the front-end
- Compatible with modern browsers (tested on Chrome and Firefox)
Result (OpenIE)

- Generated possible tuples for each sentence ranked via using TF-IDF with harmonic mean.
- Extracted canonical forms using DBpedia and FrameNet.
- Enhanced the knowledge graph.
Result (OpenIE)

Elastic DB

'The writer is a renowned Islamic Scholar and Chief Imam of Grand Muhiyuddeen Mosque of Calicut and member of All India Muslim Personal Law Board.'

OpenIE with the ranking algorithm

This sentence had the following triples as confidence level of 1.0

- Triple 1: 1.0 writer, is Scholar of, Grand Muhiyuddeen Mosque of Calicut
- Triple 2: 1.0 writer, is Islamic Scholar of, Grand Muhiyuddeen Mosque of Calicut
- Triple 3: 1.0 writer, is renowned Scholar of, Grand Muhiyuddeen Mosque
- Triple 4: 1.0 writer, is Islamic Scholar of, Grand Muhiyuddeen Mosque
- Triple 5: 1.0 writer, is renowned Islamic Scholar of, Grand Muhiyuddeen Mosque
- Triple 6: 1.0 writer, is Scholar of, Grand Muhiyuddeen Mosque
- Triple 7: 1.0 writer, is renowned Islamic Scholar of, Grand Muhiyuddeen Mosque of Calicut
- **Triple 8: 1.0 writer, is renowned Scholar of, Grand Muhiyuddeen Mosque of Calicut**

Canonical form from FrameNet & DBpedia

Subject: Writer
Relation: Fame
Object: Mosque
Export Process

- Used existing Java library to export the result obtained from OpenIE to `.gexf` file format.
- The visualization tool then imports and renders the file.
Result (Visualization)

Entity Graph

...
Neural Relation Extraction
CNN on sentences

Sentence Encoding Step

Given

- Pre-trained word vectors (New York Times dataset).
- A set of sentences and corresponding entities.

To do

- Generate a vector representation of each word in a sentence comprised of word and position embeddings.
- In the convolution layer, prepare filters that slide over a set of word vectors.
  - For example, if there are 6 words in a sentence, a filter with size 3 will slide four times.
- In the max-pooling layer, we select the maximum value from the feature map generated from the convolution layer.
CNN on sentences

Selective Attention Step

Obtained

- sentences’ representation or vector (sentence embeddings).

To do

- Have a set $S$ that contains $n$ sentences for each entity pair.
- Generate a set vector $s$ that is a weighted sum of sentence vectors.
  - Weight is computed using ‘Selective Attention’ that uses scores that tell how well the input sentence and the relation match.
- Using Softmax function, scores are determined for all relations of the set $S$. 

<table>
<thead>
<tr>
<th>Sentence 1</th>
<th>Relation 1: score (0~1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence 2</td>
<td>Relation 2: score (0~1)</td>
</tr>
<tr>
<td>Sentence 3</td>
<td>Relation 3: score (0~1)</td>
</tr>
</tbody>
</table>
Implementation details (NRE)

• Python (with Tensorflow) was used.
• Implemented RNN and CNN with Attention Layer.
• For performance measure, precision along with ROC-AUC score are used.
Result (NRE)

CNN

Precision @ 100: \textbf{0.05}
Precision @ 200: \textbf{0.05}
Precision @ 300: \textbf{0.06}
ROC-AUC score: \textbf{0.93}

Parameter used:
- number of filters (per filter size): 64
- Number of epoch: 3
- Maximum sentences per batch: 100
- Batch size: 4

30% of the training set used

RNN

Precision @ 100: \textbf{0.59}
Precision @ 200: \textbf{0.58}
Precision @ 300: \textbf{0.57}
ROC-AUC score: \textbf{0.99}

Parameter used:
- hidden units in a hidden layer: 16
- Number of epoch: 3
- Maximum sentences per batch: 100
Evaluation

- Given limited resources (i.e. Tesla K40C GPU), there was a limitation as to setting parameters for a proper experiment.
- In a limited environment, RNN works better than CNN.
Future Work

- Experiment on a better system to properly experiment the performance of RNN and CNN.
- Visualize the NRE result using the graph constructed when working with OpenIE.
- Compare the result of OpenIE and NRE quantitatively.
- Find a method to enhance the performance.
- Use the data prepared to measure the performance of NRE.
- Try other neural network models than RNN and CNN.
Conclusion

• As a part of extracting emerging knowledge from social media, relation extraction over sentences is an important task.
• OpenIE extracts relations and together with visualization does give useful information.
• NRE extracts relations given sentences and corresponding entities.
• The performance of both methods needs to be quantitatively and qualitatively analysed to see which one performs better.
Acknowledgement

I would like to express my special thanks of gratitude to my supervisor, Dongwoo Kim, who gave me guidance throughout the semester.

Furthermore, I would also like to thank researchers at ANU Computational Media Lab who gave invaluable advice.
THANK YOU