RECOGNIZING CUNEIFORM SIGNS USING GRAPH BASED METHODS

Presented By Yuval Zifrut

Based on a paper by: Nils M. Kriege, Matthias Fey, Denis Fisseler, Petra Mutzel and Frank Weichert. Department of Computer Science, TU Dortmund, Germany, Mar 2018.
TODAY OUTLINE

- Cuneiform
- Representing Cuneiform Signs By Graphs
- Graph Based Methods
- Classifying Cuneiform Graphs
- Conclusion And Future Work
The cuneiform script is one of the earliest systems of writing.
created by the Sumerians in the late 4th millennium BC.
It was in use for more than three millennia by various cultures.
These tablets are an important source for revealing early human history. They were used for simple delivery notes over religious texts or even contracts between empires. Created by impressing an angular stylus into moist clay tablets, resulting in signs consisting of tetrahedron shaped markings, called wedges.
THE EVOLUTION OF THE WRITTEN LANGUAGE

Cuneiform script
Hieroglyphs
The Latin Alphabet
Emoticons

4000 BC  2000 BC  0  2000 ACD*

*(After Cognitive Disintegration)
The method of writing was a verbal script, each letter indicating a different syllable.

For example, the wave sign can indicate a word that is so pronounced or a syllable that sounds like that within the word.

Evolution of the cuneiform sign SAG "head":

MORE ABOUT CUNEIFORM
THE DECIPHERMENT OF CUNEIFORM

- Given cuneiform tablet, how can it be deciphered?
- Bisotun Inscription, Iran - 1800
In order to separate the words, we distinguish between the different signs and see what signs are repeated many times.
Now we can know which sign combinations represent the same word and draw its meaning.
Combined with the historical background of that period (names of kings, titles, etc.), we can decipher the meaning of the signs.
Finally we can deduce how to pronounce each sign.
Over 500,000 cuneiform fragments have been discovered so far. Reading and analyzing cuneiform manuscripts is a difficult and time-consuming task, even for human experts. Tablets are digitalized with the help of high-resolution 3D measurement technologies. Automated processing of recognizing cuneiform signs is needed.

AS YOU CAN SEE...
SO, LET'S GET OVER IT!
each wedge represented by four vertices
each vertices has a labels:
vertex type - depth, tail, right, left
glyph type - vertical, horizontal, ‘Winkelhaken’
Vertices are points in
Wedge -> Tetrahedron

REPRESENTING CUNEIFORM SIGNS BY GRAPHS
The vertices of the same wedge are pairwise adjacent and form a clique.

arrangement edges - In order to obtain connected graphs we introduce additional edges between all pairs of depth vertices.
WE PRESENT TWO GRAPH BASED METHODS FOR THE CLASSIFICATION OF CUNEIFORM SIGNS

- SplineCNN - Spline graph convolutional neural network
- GED - Graph edit distance
SplineCNNs are a generalization of traditional CNNs for irregular structured data.

For a graph with vertices we store the vertex attributes in a feature matrix.

We will define some Convolution over neighboring features for a vertex.
This method uses training set.

We will use this method to get a significant improvements in accuracy.

Figure 3: Graph convolutional network architecture for classifying 30 cuneiform signs.
The relies on the following elementary operations to edit a graph:
- substitution, deletion and insertion of vertex and edge.
- Each is assigned a cost.
- A sequence of that transforms a graph into another graph is called an.
- We denote the set of from to by.
- The from to is defined by

**GRAPH EDIT DISTANCE**
computing the generalizes the classical problem, which is well known to be NP-complete.

In 2009, proposed a method to derive a series of edit operations from an optimal assignment between the vertices of the two input graphs. The method can be computed in cubic time, but not necessarily computes a minimum cost edit path from one graph to the other.

Recently, a binary linear programming formulation for computing the exact graph edit distance has been proposed (2017).

NP-COMPLETE PROBLEM
Transforming graph $G_1$ graph into $G_2$
- delete edge $e_1$ delete node $n_1$
- delete edge $e_2$ insert edge $e_4$
- insert node $n_2$ insert edge $e_3$.

Cost of edit operations:
- deletion/insertion of edges/nodes – 1,
- substitution of nodes/edges – 0.

$\text{GED}(G_1, G_2) = 6$
CHALLENGES

- On a purely geometric level without semantic context, for example, signs ‘ba’ (number 7) and ‘ku’ (24) are difficult to correctly distinguish even for human experts.
- In our words, those sings have a small distance.
We will compare two signs by the graph edit distance with a tailored cost function in order to implement a nearest neighbor classifier.
If is operations of 

- vertex substitution:
  - prevent label substitution
- edges substitution:

- vertex or edge deletion or insertion:
- D
For two cuneiform graphs $G$ and $H$ with wedges and $x$ and $y$, we create the assignment cost matrix $C$:

$$
C = \begin{bmatrix}
    c_{1,1} & c_{1,2} & \cdots & c_{1,m} & c_{1,r} & \infty & \cdots & \infty \\
    c_{2,1} & c_{2,2} & \cdots & c_{2,m} & \infty & c_{2,r} & \cdots & \cdots \\
    \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \cdots \\
    c_{n,1} & c_{n,2} & \cdots & c_{n,m} & \infty & \cdots & \cdots & c_{n,r} \\
    c_{r,1} & \infty & \cdots & \infty & 0 & 0 & \cdots & 0 \\
    \infty & c_{r,2} & \cdots & \infty & 0 & 0 & \cdots & \cdots \\
    \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \cdots \\
    \infty & \cdots & \infty & c_{r,m} & 0 & \cdots & 0 & 0 \\
\end{bmatrix},
$$

where $c_{i,j}$ denotes the cost of assigning the wedge $i$ to the wedge $j$ of the other sign, $c_{i,r}$ is the cost of deleting the wedge $i$, and $c_{r,j}$ is the cost of inserting the wedge $j$.
The following algorithm applies the above theorem to a given cost matrix to find an optimal assignment in cubic time:

- Step 1. Subtract the smallest entry in each row from all the entries of its row.
- Step 2. Subtract the smallest entry in each column from all the entries of its column.
- Step 3. Draw lines through appropriate rows and columns so that all the zero entries of the cost matrix are covered and the minimum number of such lines is used.
- Step 4. Test for Optimality: (i) If the minimum number of covering lines is n, an optimal assignment of zeros is possible and we are finished. (ii) If the minimum number of covering lines is less than n, an optimal assignment of zeros is not yet possible. In that case, proceed to Step 5.
- Step 5. Determine the smallest entry not covered by any line. Subtract this entry from each uncovered row, and then add it to each covered column. Return to Step 3.
<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alex</td>
<td>80</td>
<td>40</td>
<td>50</td>
<td>46</td>
</tr>
<tr>
<td>Ben</td>
<td>40</td>
<td>70</td>
<td>20</td>
<td>25</td>
</tr>
<tr>
<td>Charles</td>
<td>30</td>
<td>10</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Davinia</td>
<td>35</td>
<td>20</td>
<td>25</td>
<td>30</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alex</td>
<td>25</td>
<td>0</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>Ben</td>
<td>5</td>
<td>50</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jean</td>
<td>5</td>
<td>0</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>Davinia</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alex</td>
<td>24</td>
<td>0</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>Ben</td>
<td>5</td>
<td>51</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jean</td>
<td>4</td>
<td>0</td>
<td>9</td>
<td>14</td>
</tr>
<tr>
<td>Davinia</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>
The dataset for all experiments contains 267 cuneiform signs with 5,680 vertices and 23,922 edges in total.

9 cuneiform tablets written by scholars of Hittitology. Each tablet contains 30 cuneiform signs.
Table 2: Cuneiform sign classification results of all proposed methods for each test split, overall mean accuracy and standard deviation.

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>GED EXACT</td>
<td>93.24</td>
<td>96.30</td>
<td>100.0</td>
<td>96.30</td>
<td>100.0</td>
<td>92.59</td>
<td>85.19</td>
<td>85.19</td>
<td>85.19</td>
<td>100.0</td>
<td>91.67</td>
</tr>
<tr>
<td>± 5.94</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GED APX1</td>
<td>89.17</td>
<td>92.59</td>
<td>96.30</td>
<td>96.30</td>
<td>88.89</td>
<td>85.19</td>
<td>81.48</td>
<td>81.48</td>
<td>81.48</td>
<td>96.30</td>
<td>91.67</td>
</tr>
<tr>
<td>± 6.03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GED APX2</td>
<td>92.87</td>
<td>92.59</td>
<td>100.0</td>
<td>96.30</td>
<td>100.0</td>
<td>92.59</td>
<td>85.19</td>
<td>85.19</td>
<td>85.19</td>
<td>100.0</td>
<td>91.67</td>
</tr>
<tr>
<td>± 5.86</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNN without</td>
<td>87.37</td>
<td>86.56</td>
<td>86.30</td>
<td>87.41</td>
<td>90.74</td>
<td>92.60</td>
<td>93.33</td>
<td>87.04</td>
<td>79.26</td>
<td>81.48</td>
<td>90.00</td>
</tr>
<tr>
<td>± 6.16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>augmentation</td>
<td>± 4.43</td>
<td>± 3.92</td>
<td>± 5.84</td>
<td>± 6.11</td>
<td>± 4.62</td>
<td>± 2.92</td>
<td>± 3.15</td>
<td>± 5.00</td>
<td>± 3.90</td>
<td>± 4.89</td>
<td></td>
</tr>
<tr>
<td>CNN with</td>
<td>93.54</td>
<td>95.92</td>
<td>94.81</td>
<td>100.0</td>
<td>94.44</td>
<td>95.19</td>
<td>90.74</td>
<td>88.89</td>
<td>99.37</td>
<td>95.42</td>
<td></td>
</tr>
<tr>
<td>augmentation</td>
<td>± 4.40</td>
<td>± 2.73</td>
<td>± 2.59</td>
<td>± 0.00</td>
<td>± 3.15</td>
<td>± 2.92</td>
<td>± 4.00</td>
<td>± 4.62</td>
<td>± 1.91</td>
<td>± 3.65</td>
<td></td>
</tr>
</tbody>
</table>
extend the dataset by cuneiform signs that can only be distinguished in context.

FUTURE WORK
▶ Any Questions?

THE END
- The remarkable decoding of cuneiform - [https://benhateva.wordpress.com/2007/03/17/30093/](https://benhateva.wordpress.com/2007/03/17/30093/)
- Hungarian method - [http://www.math.harvard.edu/archive/20_spring_05/handouts/assignment_overheads.pdf](http://www.math.harvard.edu/archive/20_spring_05/handouts/assignment_overheads.pdf)

REFERENCES