RECOGNIZING CUNEIFORM SIGNS USING GRAPH BASED METHODS

Presented By Yuval Zifrut

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

- Cuneiform
- Representing Cuneiform Signs By Graphs
- Graph Based Methods
- Classifying Cuneiform Graphs
- Conclusion And Future Work

TODAY OUTLINE

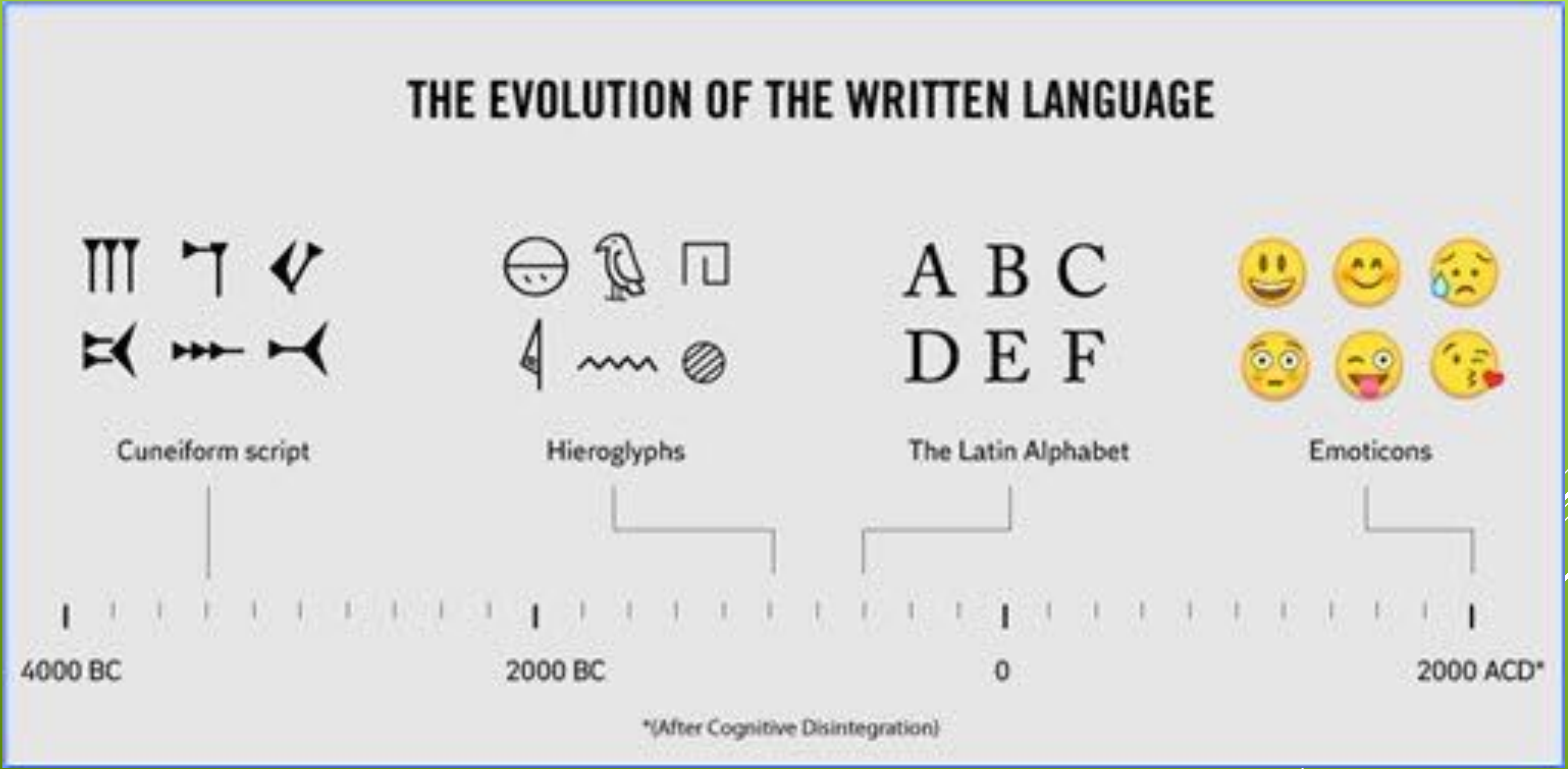
- The cuneiform script is one of the earliest systems of writing.
- created by the Sumerians in the late 4th millennium BC.
- It was in use for more than three millennia by various cultures.

INTRODUCTION

(a) Cuneiform tablet fragment

- These tablets are an important source for revealing early human history.
- They 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.

INTRODUCTION - CONTINUE


THE EVOLUTION OF THE WRITTEN LANGUAGE
A B C
D E F
Cuneiform script
Hieroglyphs
The Latin Alphabet
Emoticons
4000 BC
2000 BC
0
2000 ACD*
*(After Cognitive Disintegration)

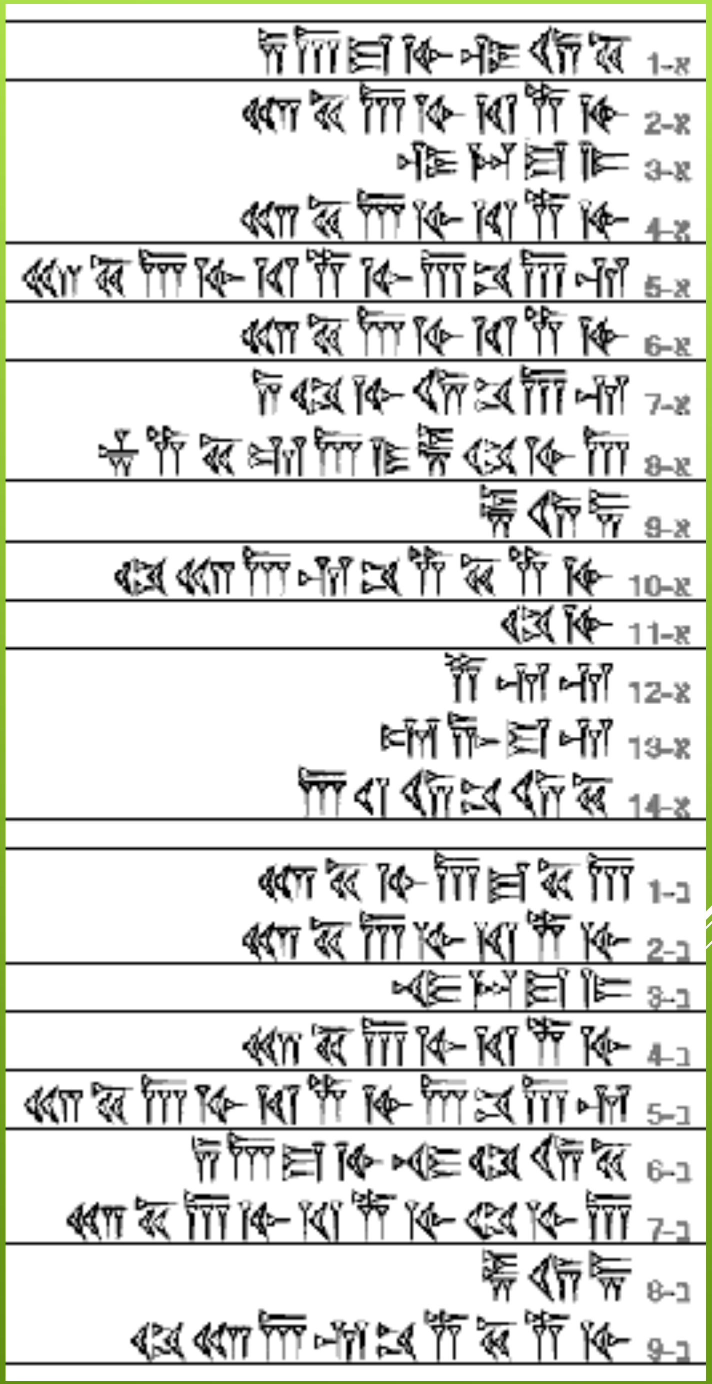
- The method of writing was a verbal script, each letter indicating a different syllable.
- For example, the wave sign can indicate a word that is so pronounced or a syllable that sounds like that within the word.
- Evolution of the cuneiform sign SAG "head“:

MORE ABOUT CUNEIFORM

- Given cuneiform tablet, how can it be deciphered?
- Bisotun Inscription, Iran - 1800

THE DECIPHERMENT OF CUNEIFORM

- In order to separate the words, we distinguish between the different signs and see what signs are repeated many times.



א-1: X
א-2: מלך
א-3: גדול
א-4: מלך
א-5: המלכים
א-6: מלך
א-7: X
א-8: X
א-9: X
א-10: X
א-11: X
א-12: X
א-13: X
א-14: X

ב-1: X
ב-2: מלך
ב-3: גדול
ב-4: מלך
ב-5: המלכים
ב-6: X
ב-7: X
ב-8: X
ב-9: X

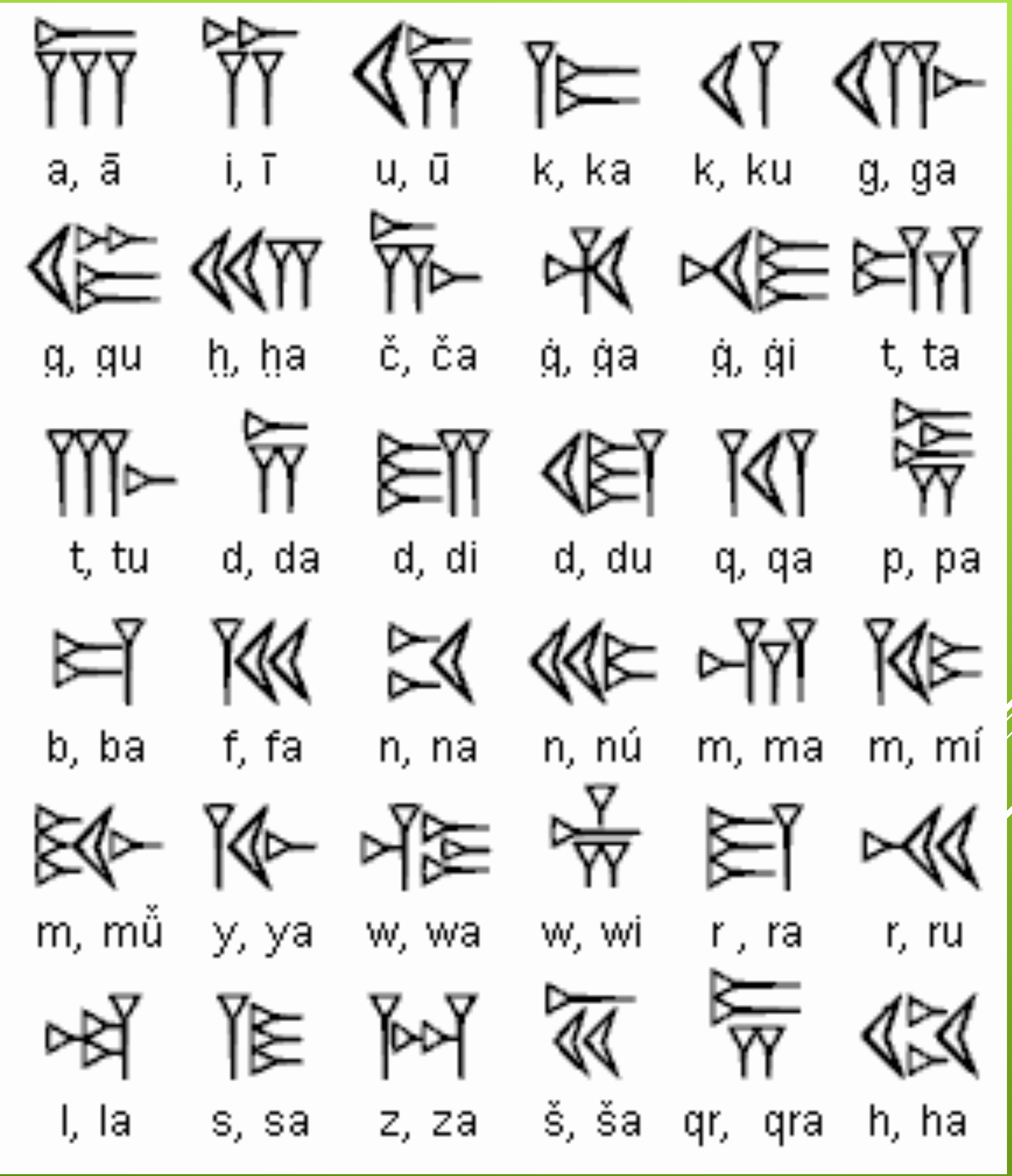
- Now we can know which sign combinations represent the same word and draw its meaning.

- Combined with the historical background of that period (names of kings, titles, etc.), we can decipher the meaning of the signs.

א-1: דריווש
א-2: מלך
א-3: גדול
א-4: מלך
א-5: המלכים
א-6: מלך
א-7: המדינות
א-9: בן
א-8: וישתספהיא
א-10: האח'מני
א-11: אשר
א-12: בנה
א-13: הארמון
א-14: הזה

ב-1: ח'שיארשא
ב-2: מלך
ב-3: גדול
ב-4: מלך
ב-5: המלכים
ב-8: בן
ב-6: דריווש
ב-7: המלך
ב-9: האח'מני

- Finally we can deduce how to pronounce each sign.

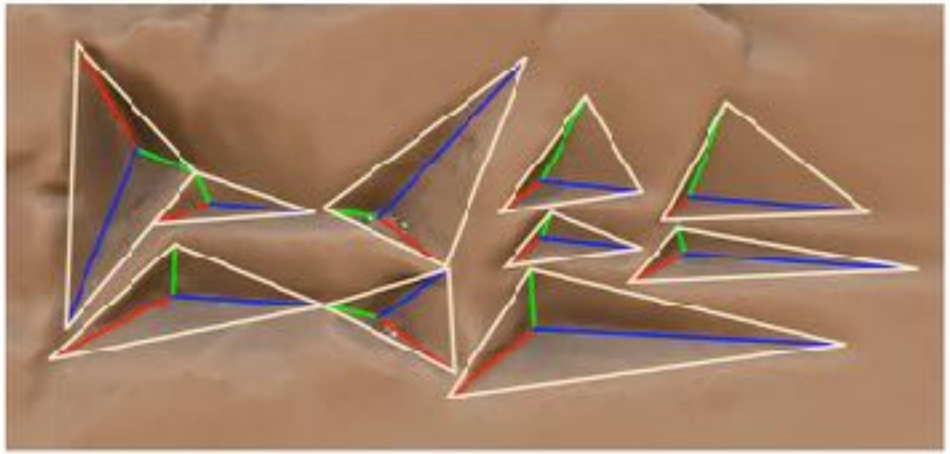


- Over 500,000 cuneiform fragments have been discovered so far
- reading and analyzing cuneiform manuscripts is difficult and time consuming task, even for human experts
- tablets are digitalized with the help of high-resolution 3D measurement technologies
- automated processing of recognizing cuneiform signs is needed

AS YOU CAN SEE…

SO, LETS 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

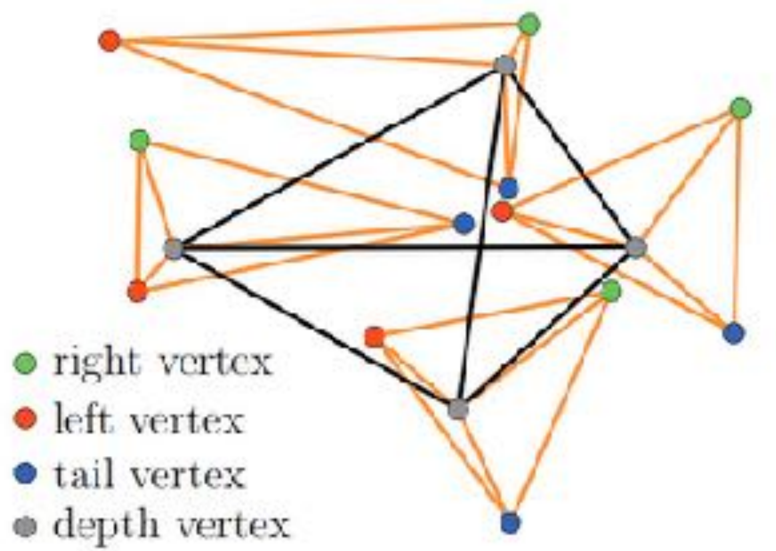


(b) Tetrahedron shaped markings

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.

REPRESENTING CUNEIFORM SIGNS BY GRAPHS - CONTINUE



(b) Graph representation

- SplineCNN - Spline graph convolutional neural network
- GED - Graph edit distance

WE PRESENT TWO GRAPH BASED METHODS FOR THE CLASSIFICATION OF CUNEIFORM SIGNS

- 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 .

SPLINE GRAPH CONVOLUTIONAL NEURAL NETWORK

- This method uses training set.
- We will use this method to get a significant improvements in accuracy.

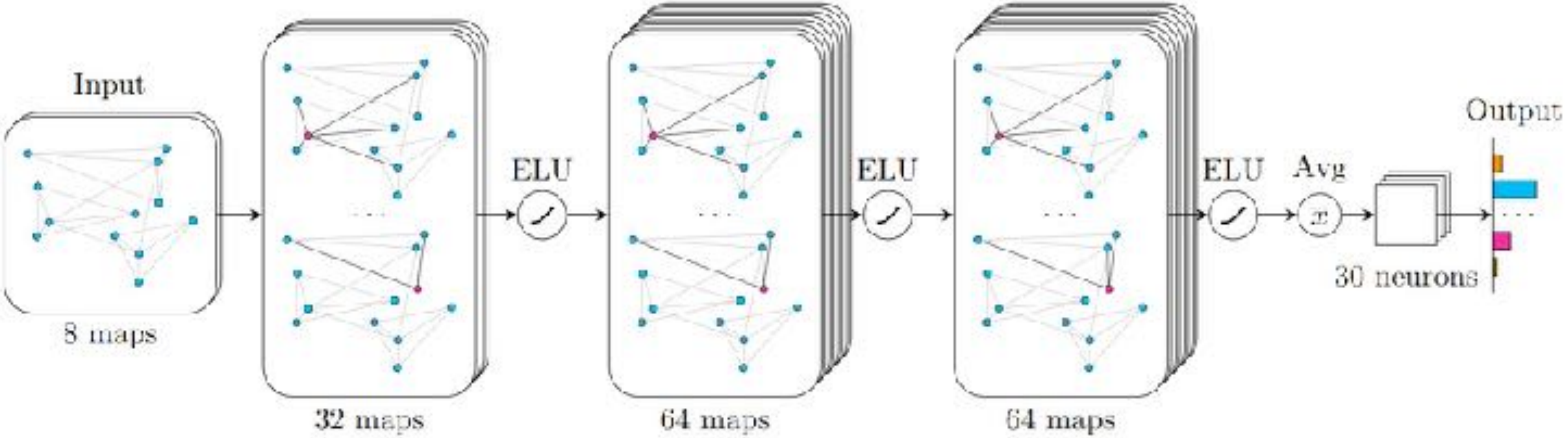


Figure 3: Graph convolutional network architecture for classifying 30 cuneiform signs.

- 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 .
- 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).

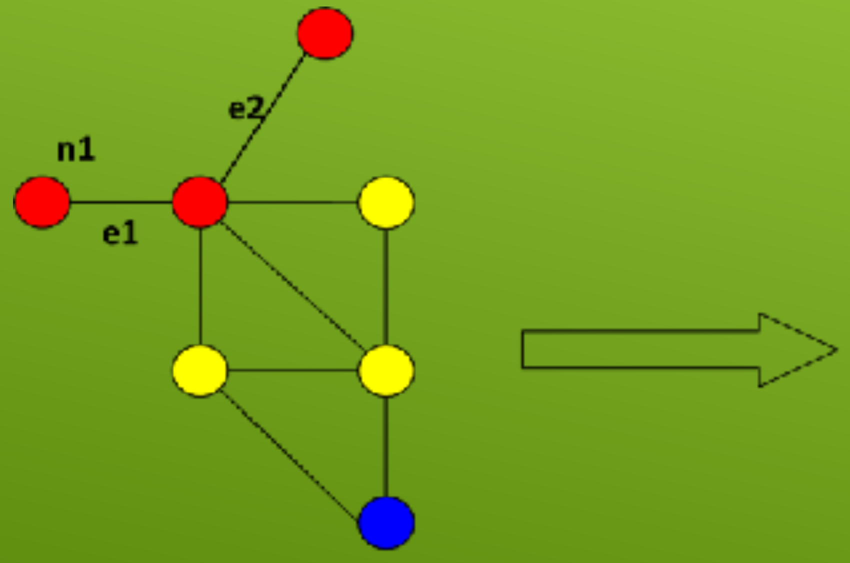
NP-COMPLETE PROBLEM

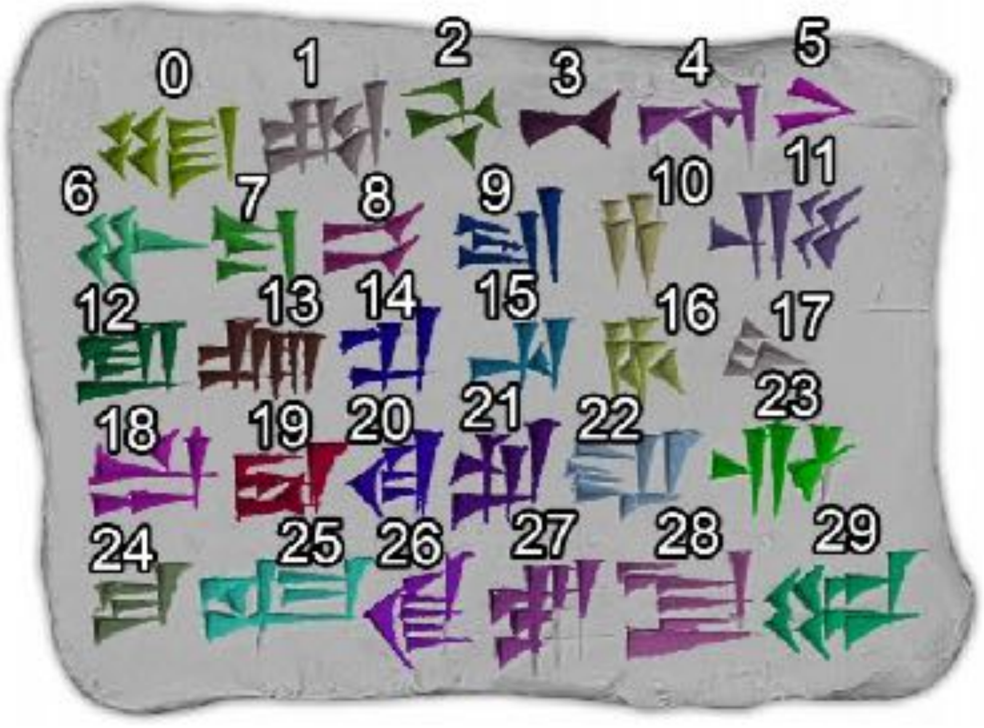
- Transforming graph **G1** graph into **G2**
  - delete edge **e1** delete node **n1**
  - delete edge **e2** insert edge **e4**
  - insert node **n2** insert edge **e3**.

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

  GED(**G1**, **G2**)=6

AN EXAMPLE





(a) Cuneiform tablet

- 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.

CHALLENGES

- We will compare two signs by the graph edit distance with a tailored cost function in order to implement a nearest neighbor classifier.

FOR OUR NEEDS

COST FUNCTION

- If is operations of
- vertex substitution :
- - prevent label substitution
- edges substitution :
- vertex or edge deletion or insertion:
- D

HEURISTICS FOR COMPUTING THE CUNEIFORM GRAPH EDIT DISTANCE

- For two cuneiform graphs and with wedges and , we create the assignment cost matrix :

$$\mathbf{C} = \left[\begin{array}{cccc|cccc} c_{1,1} & c_{1,2} & \cdots & c_{1,m} & c_{1,\epsilon} & \infty & \cdots & \infty \\ c_{2,1} & c_{2,2} & \cdots & c_{2,m} & \infty & c_{2,\epsilon} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \infty \\ c_{n,1} & c_{n,2} & \cdots & c_{n,m} & \infty & \cdots & \infty & c_{n,\epsilon} \\ \hline c_{\epsilon,1} & \infty & \cdots & \infty & 0 & 0 & \cdots & 0 \\ \infty & c_{\epsilon,2} & \ddots & \vdots & 0 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \infty & \vdots & \ddots & \ddots & 0 \\ \infty & \cdots & \infty & c_{\epsilon,m} & 0 & \cdots & 0 & 0 \end{array}\right],$$

- where denotes the cost of assigning the wedge to the wedge of the other sign, is the cost of deleting the wedge and the cost of inserting the wedge .

- 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.

THE HUNGARIAN METHOD

	1	2	3	4
Alex	80	40	50	46
Ben	40	70	20	25
Charles	30	10	20	30
Davinia	35	20	25	30

	1	2	3	4
Alex	25	0	10	1
Ben	5	50	0	0
Jean	5	0	10	15
Davinia	0	0	5	5

	1	2	3	4
Alex	25	0	10	1
Ben	5	50	0	0
Jean	5	0	10	15
Davinia	0	0	5	5

	1	2	3	4
Alex	24	0	9	0
Ben	5	51	0	0
Jean	4	0	9	14
Davinia	0	1	5	5

- 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

Class	'ba'	'bi'	'bu'	'da'	'di'	'du'	'ha'	'hi'	'hu'	'ka'	'ki'	'ku'	'la'	'li'	'lu'
$\#G$	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
$\|V\|$	36	24	16	8	16	8	20	16	16	20	16	28	24	28	16
$\|E\|$	180	102	60	26	60	26	80	60	60	80	60	126	102	126	60
h_{max}	7.5	7.4	10.0	5.3	7.5	7.7	7.3	7.9	7.5	9.1	9.1	7.4	7.7	6.7	7.5
w_{max}	12.7	11.7	13.4	7.3	14.7	8.7	13.9	10.0	9.4	11.1	5.8	13.7	11.4	10.6	15.0

Class	'na'	'ni'	'nu'	'ra'	'ri'	'ru'	'sa'	'si'	'su'	'ta'	'ti'	'tu'	'za'	'zi'	'zu'
$\#G$	9	9	9	9	9	9	9	9	9	9	9	9	8	8	8
$\|V\|$	16	24	16	20	20	20	20	24	20	16	28	28	24	28	36
$\|E\|$	60	102	60	80	80	80	80	102	80	60	126	126	102	126	180
h_{max}	8.5	9.9	10.2	7.7	8.9	8.2	10.0	9.2	8.2	8.3	12.6	10.5	9.1	8.9	10.3
w_{max}	14.4	7.6	8.6	9.6	11.5	10.2	10.4	11.2	11.7	8.4	15.2	12.4	9.2	13.2	21.0

Table 2: Cuneiform sign classification results of all proposed methods for each test split, overall mean accuracy and standard deviation.

		# Split									
Method	Mean	1	2	3	4	5	6	7	8	9	10
GED EXACT	93.24 ± 5.94	**96.30**	**100.0**	96.30	**100.0**	92.59	85.19	85.19	85.19	**100.0**	91.67
GED APX1	89.17 ± 6.03	92.59	96.30	96.30	88.89	85.19	81.48	81.48	81.48	96.30	91.67
GED APX2	92.87 ± 5.86	92.59	**100.0**	96.30	**100.0**	92.59	85.19	85.19	85.19	**100.0**	91.67
CNN without augmentation	87.37 ± 6.16	86.56 ± 4.43	86.30 ± 3.92	87.41 ± 5.84	90.74 ± 6.11	92.60 ± 4.62	**93.33** ± 2.92	87.04 ± 3.15	79.26 ± 5.00	81.48 ± 3.90	90.00 ± 4.89
CNN with augmentation	**93.54** ± 4.40	95.92 ± 2.73	94.81 ± 2.59	**100.0** ± 0.00	94.44 ± 3.15	**95.19** ± 1.79	89.63 ± 2.92	**90.74** ± 4.00	**88.89** ± 4.62	99.37 ± 1.91	**95.42** ± 3.65

FUTURE WORK

- extend the dataset by cuneiform signs that can only be distinguished in context.

- Any Questions?

THE END

REFERENCES

- Recognizing Cuneiform Signs Using Graph Based Methods - https://arxiv.org/pdf/1802.05908.pdf
- Cuneiform script - https://en.wikipedia.org/wiki/Cuneiform_script
- Behistun Inscription - https://en.wikipedia.org/wiki/Behistun_Inscription
- The remarkable decoding of cuneiform - https://benhateva.wordpress.com/2007/03/17/30093/
- SplineCNN - https://www.groundai.com/project/splinecnn-fast-geometric-deep-learning-with-continuous-b-spline-kernels/
- GED – http://gedevo.mpi-inf.mpg.de/
- Hungarian method - http://www.math.harvard.edu/archive/20_spring_05/handouts/assignment_overheads.pdf
- Video - https://www.khanacademy.org/humanities/ancient-art-civilizations/ancient-near-east1/the-ancient-near-east-an-introduction/v/cuneiform