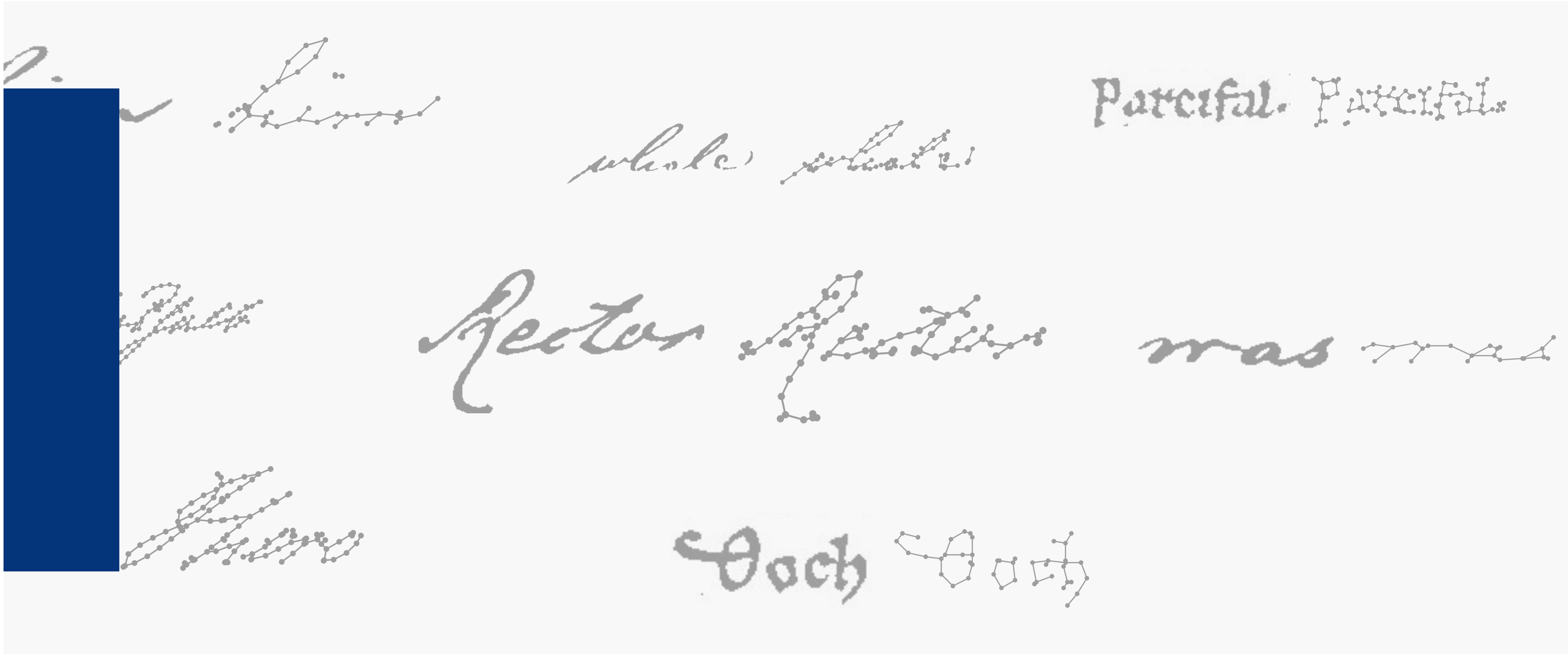
DAS 2018

Graph-Based Keyword Spotting in Historical Documents Using Context-Aware Hausdorff Edit Distance

Michael Stauffer, Andreas Fischer, and Kaspar Riesen

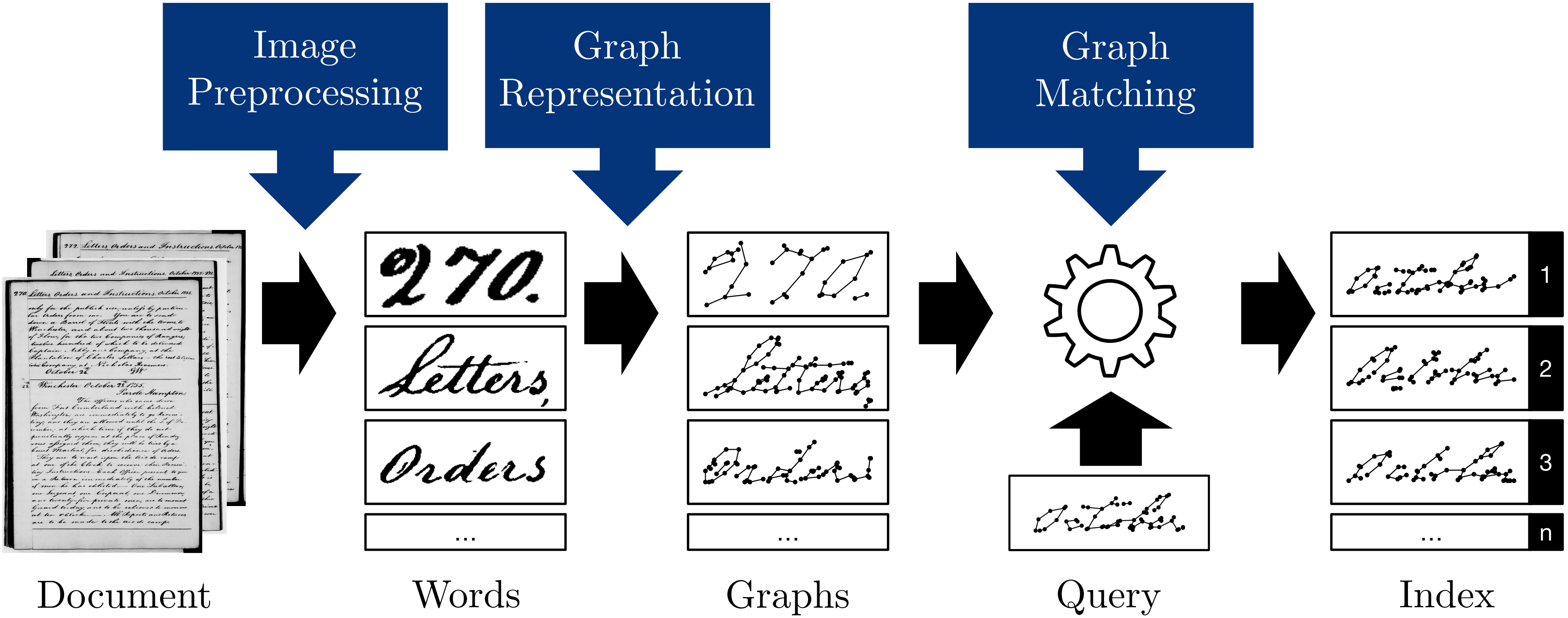


Content

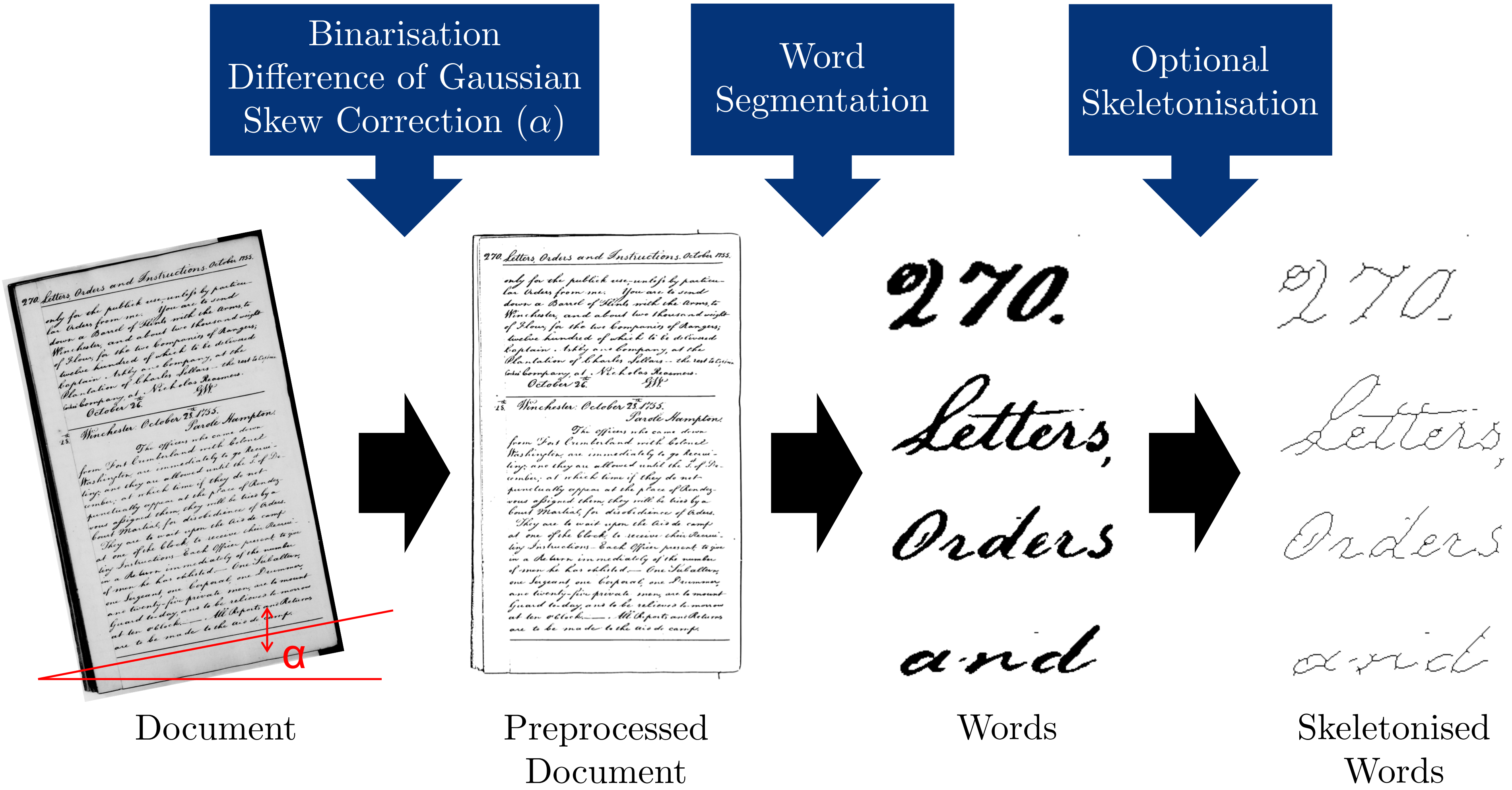
- Graph-based Keyword Spotting
  - Image Preprocessing
  - Graph Representation
  - Graph Matching
    - Graph Edit Distance
    - Context-Aware Hausdorff Edit Distance
    - Ensemble Methods
- Experiments
- Conclusion
- Q+A

Graph-based Keyword Spotting (KWS) – Overview

Graph-based KWS is based on the **representation of words** by means of different graphs. This representations are eventually used to **retrieve a keyword** by **matching a query** graph **with** all **document graphs**.



Graph-based Keyword Spotting – Image Preprocessing

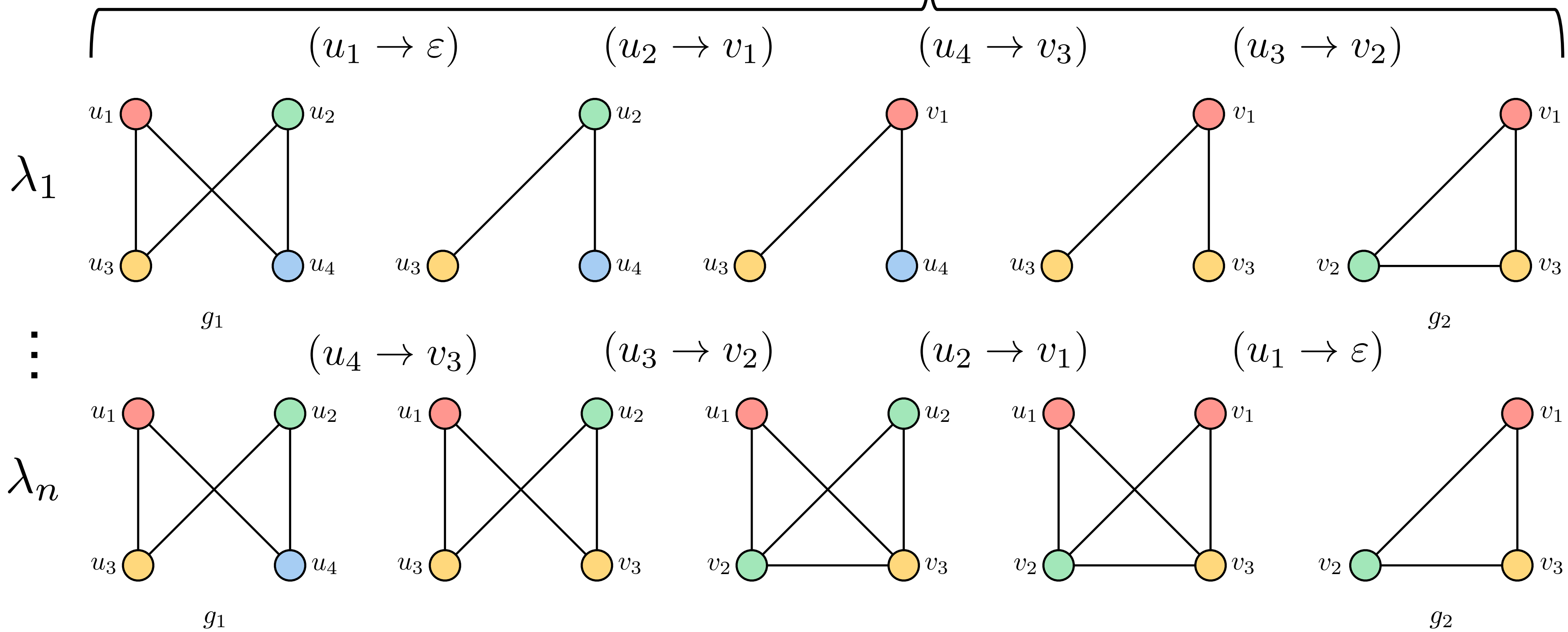


Graph-based Keyword Spotting – Graph Representation

Graph-based Keyword Spotting – Graph Edit Distance (GED)

$$d_{\text{GED}}(g_1, g_2) = \min_{\lambda \in \Upsilon(g_1, g_2)} \sum_{e_i \in \lambda} c(e_i)$$

Set of all complete edit paths

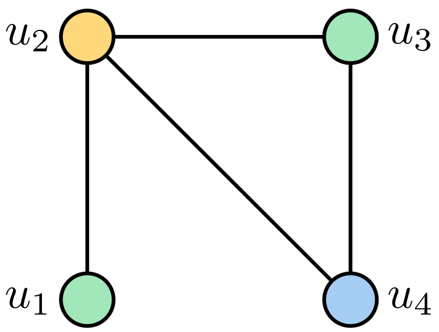


Graph-based Keyword Spotting – Hausdorff Edit Distance (HED)

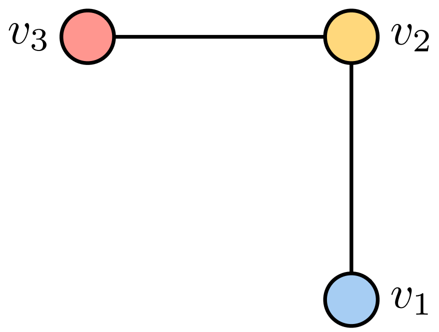
The number of edit path is exponential, and thus, HED reduces the problem of GED to a set matching problem between local substructures.

$$d_{\mathrm{HED}}(g_1, g_2) = \sum_{u \in V_1} \min_{v \in V_2 \cup \{\epsilon\}} h(u, v) + \sum_{v \in V_2} \min_{u \in V_1 \cup \{\epsilon\}} h(u, v)$$

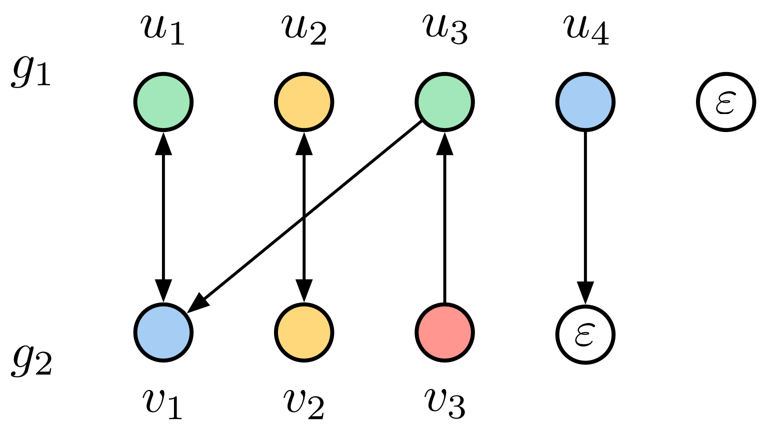
where $h(u, v)$ is the assignment cost of nodes and adjacent edges.



g_1



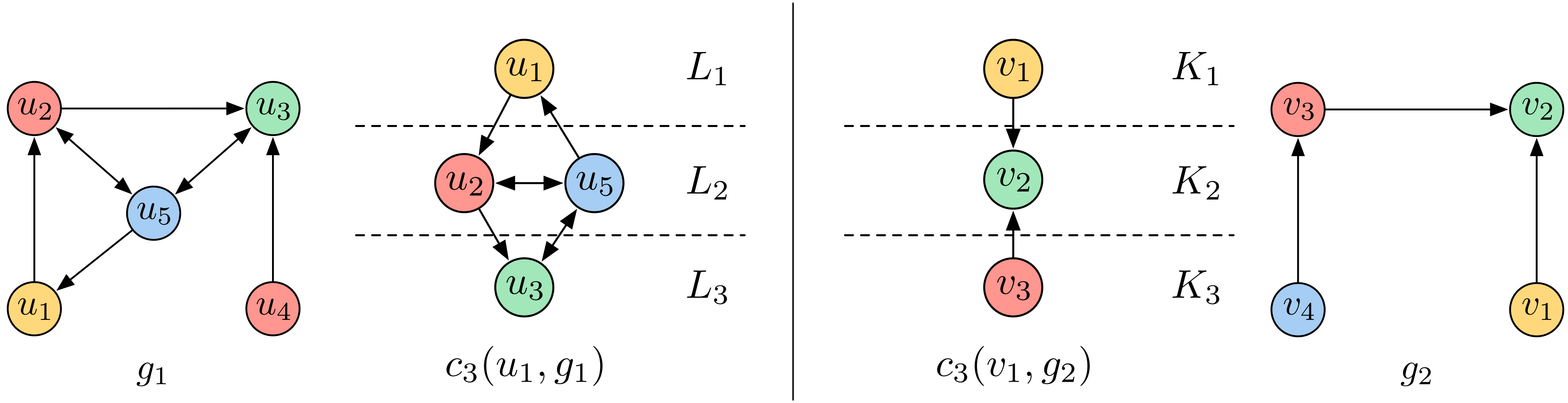
g_2



Graph-based Keyword Spotting – Context-Aware Hausdorff Edit Distance (CED)

How dissimilar/similar is graph context $c_3(u_1, g_1)$ and $c_3(v_1, g_2)$?

$$d(c_n(u, g_1), c_n(v, g_2)) = \sum_{i=1}^{n} d_{HED}(L_i, K_i)$$



$$d_{\mathrm{CED}}(g_1, g_2, n) \sum_{u \in V_1} \min_{v \in \{b_n(u,g_2),\epsilon\}} h(u,v) + \sum_{v \in V_2} \min_{u \in \{b_n(v,g_1),\epsilon\}} h(u,v) \quad ,$$

where b_n is the node in g_1/g_2 with minimum structural node context distance.

Graph-based Keyword Spotting – Ensemble Methods

mean-Graph(n)

$n = \{2, 4, 6\}$

$m =$ Keypoint (K)	$m =$ Projection (P)
0.756	0.831

→ mean → 0.794

mean-Context(m)

$m = \{$K,P$\}$

$n = 2$	$n = 4$	$n = 6$
0.756	0.673	0.678

→ mean → 0.702

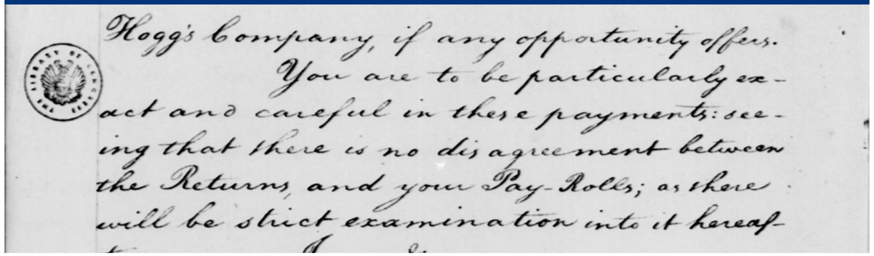
mean-All

$m =$ K, $n = 2$	$m =$ K, $n = 4$	$m =$ K, $n = 6$
0.756	0.673	0.678
$m =$ P, $n = 2$	$m =$ P, $n = 4$	$m =$ P, $n = 6$
0.831	0.563	0.456

→ mean → 0.660

Experiments – Datasets

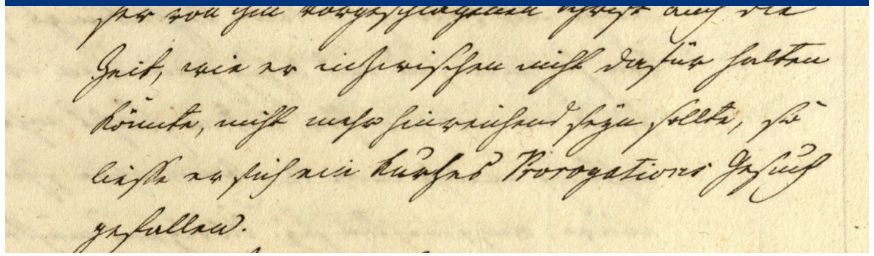
George Washington (GW)


Hogg's Company, if any opportunity offers.
You are to be particularly ex-
act and careful in these payments: see-
ing that there is no disagreement between
the Returns and your Pay-Rolls; as there
will be strict examination into it hereaf-

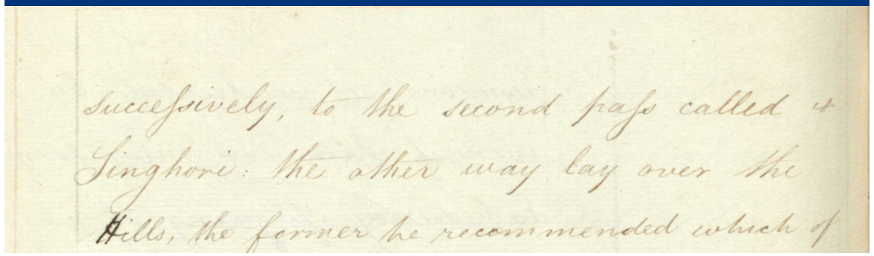
Parzival (PAR)

Graph-based vs. template-based KWS

Alvermann Konzilsprotokolle (AK)



Botany (BOT)


successively, to the second pass called +
Singhore: the other way lay over the
Hills, the former he recommended which of

Graph-based vs. learning-based KWS

Experiments – Results Graph-based KWS Methods (Accuracy)

Bipartite Graph Edit Distance (BP) vs. Hausdorff Edit Distance (HED) and Context-Aware Hausdorff Edit Distance (CED).

Method		GW		PAR		BOT		AK	
		MAP	±	MAP	±	MAP	±	MAP	±
BP	`Keypoint`	66.08		62.04		45.06		77.24	
	`Projection`	61.43		66.23		49.57		76.02	
HED	`Keypoint`	69.28	+3.19	69.23	+7.19	51.74	+6.68	79.72	+2.48
	`Projection`	66.71	+5.28	72.82	+6.59	51.69	+2.12	81.06	+5.04
CED ($n = 2$)	`Keypoint`	**71.23**	+5.15	68.39	+6.35	51.59	+6.53	80.73	+3.49
	`Projection`	67.04	+5.62	**74.10**	+7.87	51.98	+2.41	80.28	+4.26
CED ($n = 4$)	`Keypoint`	68.46	+2.37	68.64	+6.60	52.18	+7.12	81.43	+4.19
	`Projection`	64.51	+3.08	73.99	+7.76	52.02	+2.45	82.42	+6.40
CED ($n = 6$)	`Keypoint`	65.23	−0.85	67.19	+5.15	**52.45**	+7.39	**82.69**	+5.45
	`Projection`	60.25	−1.18	72.45	+6.22	50.59	+1.02	81.58	+5.56

Accuracy is measured by Mean Average Precision

Experiments – Results Graph-based KWS Methods (Time)

Average matching time (ms) per graph pair using BP with cubic time complexity, and HED and CED with quadratic time complexity.

Method	GW	Speed-Up	PAR	Speed-Up	Average	Speed-Up
BP	303.0		437.5		370.3	
HED	**3.2**	x 95	**5.2**	x 84	**4.2**	x 88
CED ($n = 2$)	7.5	x 40	12.5	x 35	10.0	x 37
CED ($n = 4$)	14.6	x 21	20.1	x 22	17.4	x 21
CED ($n = 6$)	17.5	x 17	30.4	x 14	23.9	x 15

Experiments – Results Graph-based vs. Template-based KWS Methods

Graph-based ensemble methods are able to outperform state-of-the-art template-based reference systems.

Method		GW		PAR		Average	
Reference (Template)							
- DTW'01	Geometric (Marti)	45.26		46.78		46.02	
- DTW'08	Histogram of Oriented Gradient	63.39		47.52		55.46	
- DTW'09	Histogram of Oriented Gradient	64.80		73.49		69.15	
- DTW'16	Deep Learning (unsupervised)	68.64		72.38		70.51	
Graph (Ensemble)							
- mean-Graph ($n = 2$)		**74.62**	(1)	80.06	(2)	**77.34**	(1)
- mean-Graph ($n = 4$)		73.22	(3)	79.79	(3)	76.50	(3)
- mean-Graph ($n = 6$)		69.67		79.11		74.39	
- mean-Context ($m = \mathtt{K}$)		74.27	(2)	77.36		75.82	
- mean-Context ($m = \mathtt{P}$)		67.35		78.66		73.00	
- mean-All		73.06		**81.08**	(1)	77.07	(2)

Accuracy is measured by Mean Average Precision

Experiments – Results Graph-based vs. Learning-based KWS Methods

Graph-based ensemble methods are able to keep up with state-of-the-art learning-based reference systems.

Method		BOT		AK		Average	
Reference (Learning)							
- SVM	Pyramidal Histogram of Characters	75.77	(2)	77.91		76.84	
- CNN (PHOCNet)	Pyramidal Histogram of Characters	**89.69**	(1)	**96.05**	(1)	**92.87**	(1)
- CNN	Discrete Cosine Transform of Words	54.95		82.15		68.55	
Graph (Ensemble)							
- mean-Graph ($n = 2$)		71.83		81.52		76.68	
- mean-Graph ($n = 4$)		72.43		84.22		78.33	(2)
- mean-Graph ($n = 6$)		69.17		85.20	(3)	77.19	
- mean-Context ($m = \texttt{K}$)		72.51	(3)	83.42		77.97	
- mean-Context ($m = \texttt{P}$)		67.60		85.46	(2)	76.53	
- mean-All		71.67		84.49		78.08	(3)

Accuracy is measured by Mean Average Precision

Conclusion

Graph-based KWS using Context-Aware Hausdorff Edit Distance

improves

Accuracy
(Structural Node Context)

Matching Time
(Quadratic Time Complexity)

Q+A

?!