Hyperlink Classification via Structured Graph Embedding

Geon Lee[1], Seonggoo Kang[2] and Joyce Jiyoung Whang*[1]

[1] Sungkyunkwan Univeristy (SKKU), [2] Naver Corporation, * Corresponding Author

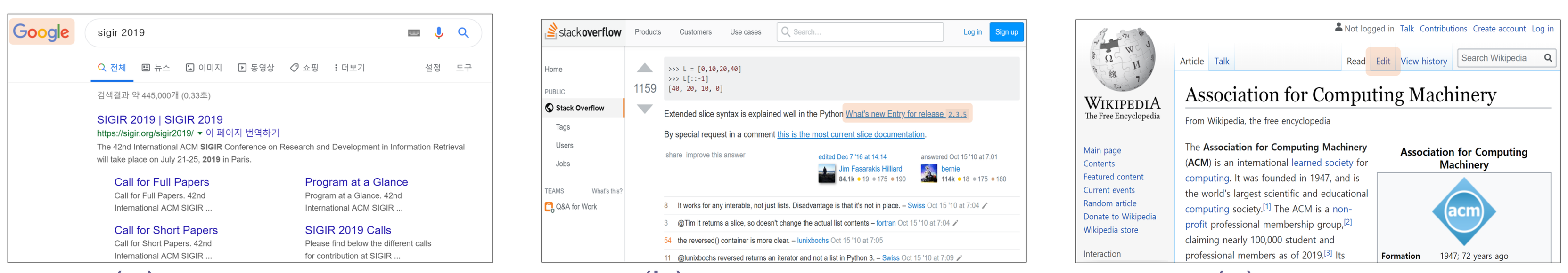
The 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019

Main Contributions

- We formally define a hyperlink classification problem in web search by classifying hyperlinks into three classes based on their roles: navigation, suggestion, and action.
- We approach the problem from a structured graph embedding perspective, by modifying knowledge graph embedding techniques.
- Relation perturbation in negative sampling enables us to significantly improve performance in classifying hyperlinks on web graphs.

Real-World Web Graphs

- The hyperlinks are created for different reasons, and play different roles.
  - Navigation links are designed to navigate the main website.
  - Suggestion links suggest users to take a look at related information.
  - Action links are made to invoke actions such as 'edit', or 'send an email'.



(a) Navigation link (b) Suggestion link (c) Action link

- We create three real-world web graphs by crawling a set of web pages and the hyperlinks starting from a web page in Stack Overflow.

	$\|V\|$	$\|E\|$	*navigation*	*suggestion*	*action*
`web_437`	404	437	268 (61.33%)	112 (25.63%)	57 (13.04%)
`web_1442`	332	1,442	1,284 (89.04%)	93 (6.45%)	65 (4.51%)
`web_10000`	2,202	10,000	9,892 (98.92%)	85 (0.85%)	23 (0.23%)

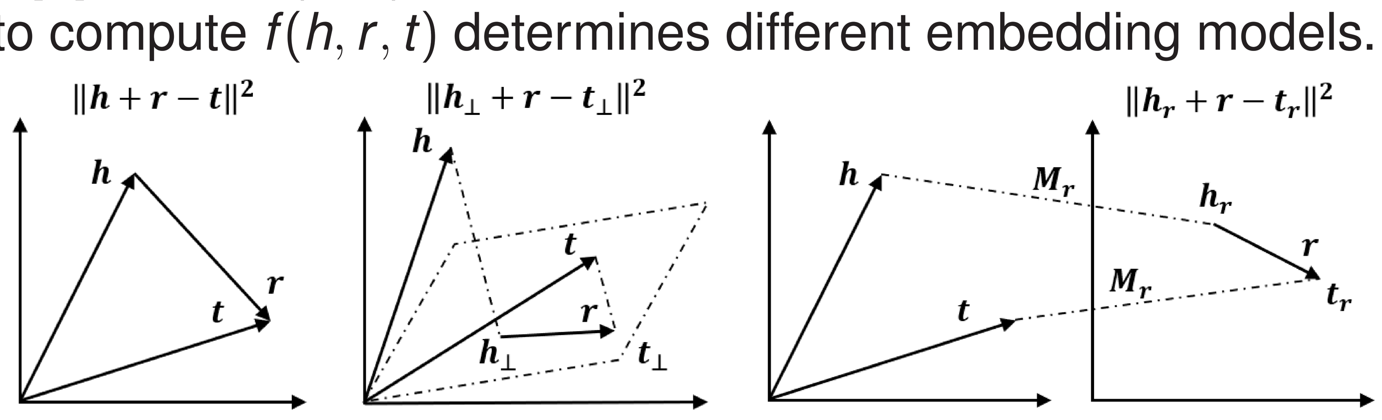
All the datasets/codes are available on **http://bigdata.cs.skku.edu**.

Knowledge Graph Embedding

- A knowledge graph is a graphical representation of human knowledge.
  - Each fact can be described as a triplet (*head entity, relation, tail entity*).
- The goal of knowledge graph embedding is to represent entities and relations in a feature space while preserving the structure of the graph.
  - Given a set of golden triplets (denoted by S) and a set of corrupted triplets (denoted by S'), minimize the following loss function:

$$L = \sum_{(h,r,t)\in\mathcal{S}} \sum_{(h',r,t')\in\mathcal{S}'} [f(h,r,t) + \gamma - f(h',r,t')]_+$$

  where $[x]_+ \equiv max(0, x)$ and γ is the margin.
  - How to compute $f(h, r, t)$ determines different embedding models.



(a) TransE (b) TransH (c) TransR

Image from "Knowledge graph embedding: A survey of approaches and applications." TKDE 2017.

Hyperlink Classification Model

- Given a directed web graph $G = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{p_1, p_2, \cdots, p_n\}$ and $\mathcal{E} = \{(p_i, p_j) : p_i \in \mathcal{V}, p_j \in \mathcal{V}\}$, each hyperlink r belongs to one of the three relation labels $\mathcal{R} = \{n, s, a\}$.
- Given a golden triplet (p_i, r, p_j), generate a corrupted triplet $c(p_i, r, p_j)$.
  - Minimize the following loss function.

$$L = \sum_{(p_i,r,p_j)\in\mathcal{S}} [f(p_i, r, p_j) + \gamma - f(c(p_i, r, p_j))]_+$$

  - TransE, TransH, and TransR only corrupt entities.
  - We corrupt an entity with probability α, and corrupt the relation with probability $1-\alpha$ $(0 < \alpha \le 1)$.

$$c(p_i,r,p_j) = \begin{cases} \text{prob. } \alpha/2: & (p_i,r,q), q \in \mathcal{V}\backslash\{p_j\}, (p_i,r,q) \notin \mathcal{S} \\ \text{prob. } \alpha/2: & (q,r,p_j), q \in \mathcal{V}\backslash\{p_i\}, (q,r,p_j) \notin \mathcal{S} \\ \text{prob. } (1-\alpha): & (p_i,r',p_j), r' \in \mathcal{R}\backslash\{r\} \end{cases}$$

- For a directed edge (p_i, p_j), we predict the relation r for (p_i, p_j) by computing

$$r^* = \underset{r\in\mathcal{R}}{\text{argmin}}\, f(p_i, r, p_j)$$

  where r^* is the predicted relation.

Hyperlink Classification Model (Cont'd)

- False negative: when we corrupt entities, there is a chance that it is not a *corrupted* one but just *unobserved* one in the train set.
  - If we corrupt a golden triplet (p_1, n, p_2) to (p_1, n, p_3), there is a risk that (p_1, n, p_3) does not exist in the train set, but exist in the valid or test sets.
  - The navigation links are prevalent while there are very few suggestion and action links. This bias makes the entity corruption undesirable.
- If we corrupt a relation, it is guaranteed that the corrupted triplet is not in the test set because each pair of web pages has a unique relation.
  - If (p_1, n, p_2) is observed, (p_1, s, p_2) or (p_1, a, p_2) should not hold.
- If we only corrupt relations and do not corrupt entities to create the negative triplets, we might have an overfitting problem and the model is not sufficiently trained for an unobserved entity.

Experimental Results

- The average F1 of our model with different α values.

		`TransE`	`TransH`	`TransR`
`web_437`	Our model, $\alpha = 0.3$	34.29	**60.25**	57.99
	Our model, $\alpha = 0.5$	34.39	58.87	57.32
	Our model, $\alpha = 0.7$	33.88	58.91	**59.83**
	The original model	**36.22**	54.04	53.22
`web_1442`	Our model, $\alpha = 0.3$	23.39	53.42	**50.04**
	Our model, $\alpha = 0.5$	**24.86**	**55.16**	46.18
	Our model, $\alpha = 0.7$	21.18	52.70	45.12
	The original model	20.05	29.94	10.35
`web_10000`	Our model, $\alpha = 0.3$	**20.68**	**76.00**	**53.86**
	Our model, $\alpha = 0.5$	17.98	74.64	46.99
	Our model, $\alpha = 0.7$	19.50	72.94	44.11
	The original model	15.31	25.35	2.08

- F1 score of each class, and the average F1 score.
  - Random-predict: random prediction while preserving the number of hyperlinks in each class.
  - Rule-based: consider within-domain hyperlinks to be navigation links, the hyperlinks associated with an anchor text containing 'edit', 'share', 'email' or 'vote' to be action links, and the rest to be suggestion links.

		navigation	*suggestion*	*action*	Average
`web_437`	Random-predict	59.75	25.81	11.07	32.21
	Rule-based	60.20	20.96	0.00	27.05
	`TransE`-original	55.78	31.96	20.93	36.22
	`TransH`-original	70.80	52.75	38.56	54.04
	`TransR`-original	67.87	52.86	38.94	53.22
	Our Model	**77.04**	**57.05**	**46.64**	**60.25**
`web_1442`	Random-predict	89.13	5.18	5.65	33.32
	Rule-based	72.98	10.20	36.67	39.95
	`TransE`-original	42.54	8.57	9.05	20.05
	`TransH`-original	54.80	13.57	21.45	29.94
	`TransR`-original	0.00	12.97	18.09	10.35
	Our Model	**93.48**	**22.88**	**49.12**	**55.16**
`web_10000`	Random-predict	98.91	1.60	0.00	33.50
	Rule-based	68.81	1.74	9.92	26.82
	`TransE`-original	43.25	2.06	0.61	15.31
	`TransH`-original	63.01	12.02	1.03	25.35
	`TransR`-original	0.00	5.61	0.61	2.08
	Our Model	**99.66**	**83.22**	**45.12**	**76.00**

- Comparison with randomly shuffled graphs where the relation labels are randomly shuffled while preserving the number of hyperlinks in each class.
  - A web graph preserves a characterized structure with respect to the three different types of hyperlinks.

		navigation	*suggestion*	*action*
`web_437`	Original Graph	77.04	57.05	46.64
	Randomly Shuffled Graph	58.60	25.36	13.79
`web_1442`	Original Graph	93.48	22.88	49.12
	Randomly Shuffled Graph	86.08	6.19	5.68
`web_10000`	Original Graph	99.66	83.22	45.12
	Randomly Shuffled Graph	98.43	1.28	0.61

Conclusion & Future Work

- By introducing an effective relation perturbation in embedding models, we can successfully classify hyperlinks on web graphs.
- We plan to extend our analysis to a case where we can incorporate various features or attributes of web pages or hyperlinks.