GBMTab: A Graph-Based Method for Interpreting Semantic Table to Knowledge Graph

Lianzheng Yang[1], Shuyang Shen[2], Jingyi Ding[3], and Jiahui Jin[3]

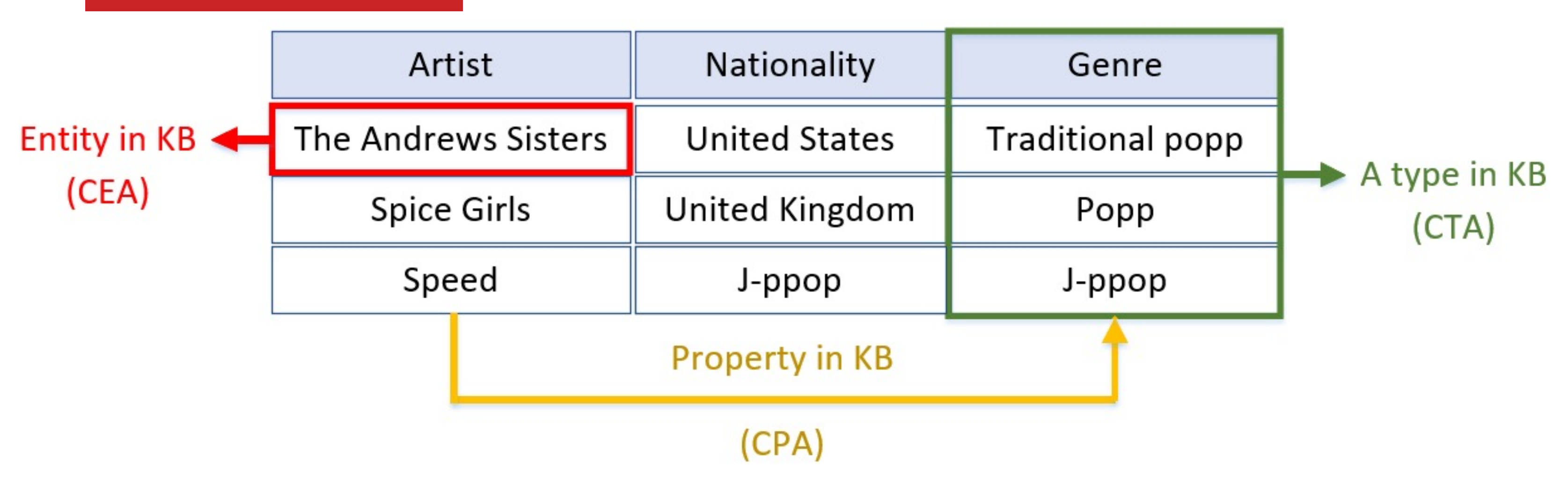
[1]School of Cyber Science and Engineering, Southeast University, China

[2]Chien-Shiung Wu College, Southeast University, China

[3]School of Computer Science and Engineering, Southeast University, China

jjin@seu.edu.cn

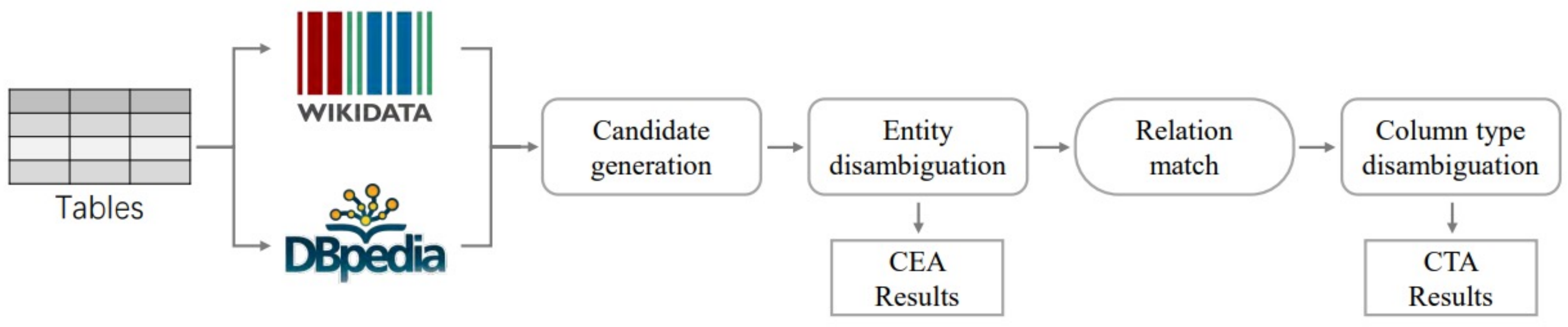
Introduction



Three sub-tasks of SemTab 2021.

- **Background:** Tabular data on the Web contains rich semantic information, so matching the tabular data into knowledge bases is an important problem.
- **Challenge:** It is challenging to interpret semantic tabular data due to the diversity of languages and noise mentions.
- **Methods:** We proposed a semantic table interpretation framework called GBMTab to solve Cell Entity Annotation (CEA) and Column Type Annotation (CTA) tasks by using a probability graph model and a knowledge graph path-matching method.

Framework



Method of CEA

- **Candidate generation**

DBpedia

➢ **String similarity comparison**: Define **s** as a mention and **e** as an entity.

$$StringSimilarity(s,e) = 1 - \frac{LevenshteinDistance(s,e)}{\max\{length(s), length(e)\}}$$

➢ **Noise mentions repair:** Use Google search engine to correct noise mentions (e.g. "cat" becomes "catt").

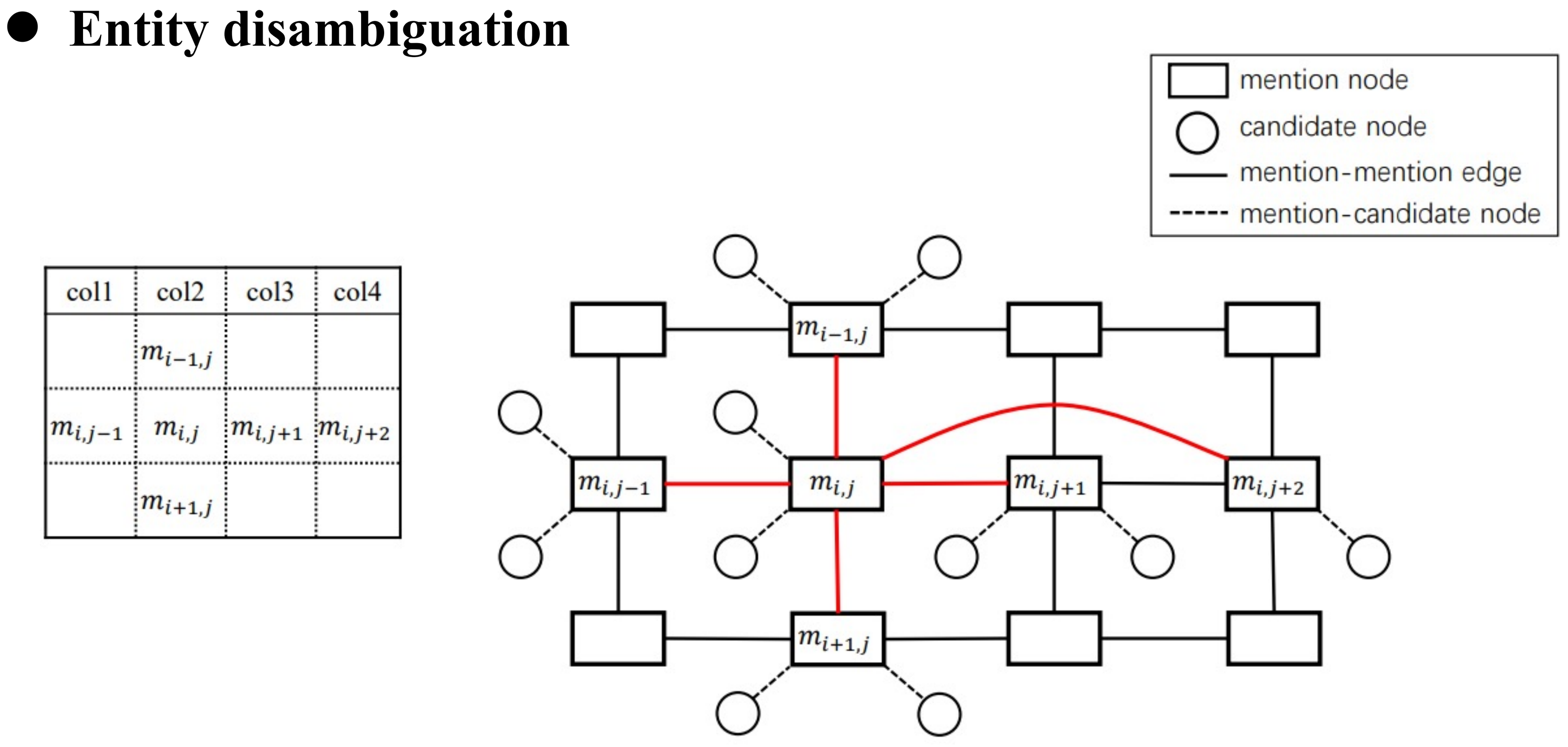
➢ **Multilingual:** Introduce multilingual DBpedia datasets.

Wikidata

➢ **Wikidata MediaWiki API:** Query MediaWiki API by posting a mention and setting the limits to a maximum of 50.

➢ **Correction of noise mentions:** Same as DBpedia.

- **Entity disambiguation**



The flowchart of entity disambiguation

➢ **Build disambiguation graph:** Starting with a given mention ($m_{i,j}$), create a disambiguation graph of all other mentions in the same row or column and mention's ($m_{i,j}$) corresponding candidates.

➢ **Build features between nodes:**

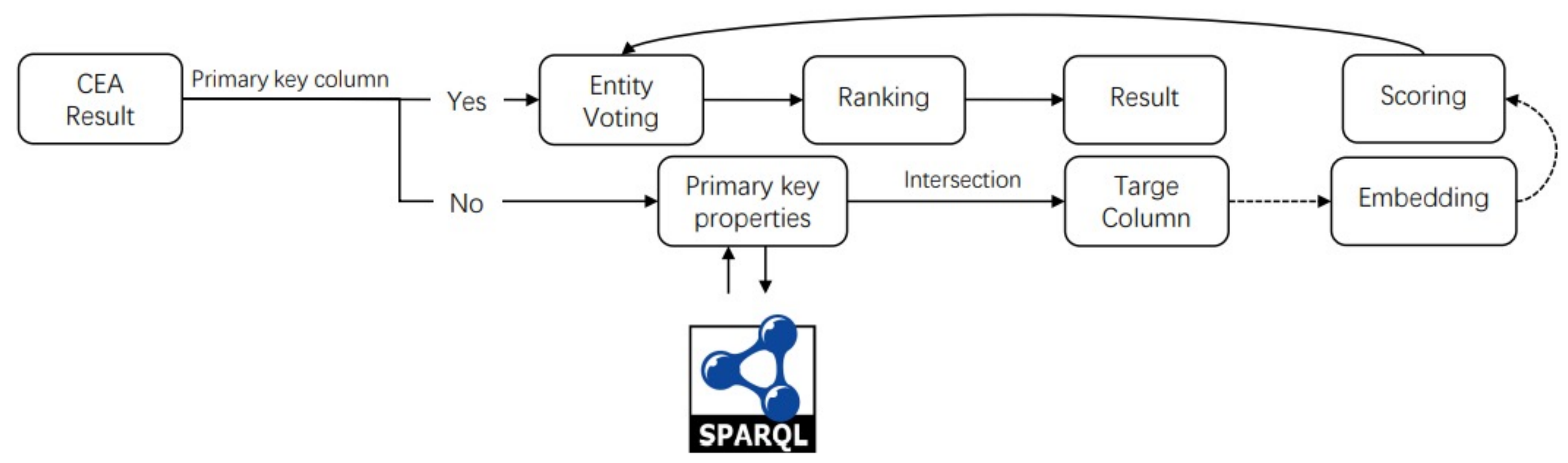
Priori Features: Calculate priori features from Knowledge Base and WDC[1].

Context Features: Take the values of cells in the same row or column of objective cell as its feature, and use Levenshtein distance and cosine distance to rank the candidate entities.

Abstract Features: Intersect abstract of an entity with the other available text features and score it with cosine distance.

➢ **Iterative probability propagation:** Greedily assigns the current value of a node to its maximum likelihood value, continuously calculates and updates the feature of the mention, and finally reaches the global optimal solution.

Method of CTA



The flowchart of column type annotation.

- **Relation match:** We use the Wikidata SPARQL endpoint to search for the type of each entity in the same column.
- **Column type disambiguation:** If the intersection of properties and search candidates is empty, we will use a hybrid method which consists of vote mechanism, embedding distance and text similarity to rank types in candidate set. We also use a knowledge graph path-matching method to choose the most suitable relation path for those entities whose attribute values match.

Result

	Round1			Round2		
	F1	Precision	Rank	F1	Precision	Rank
CEA	0.692	0.692	2	0.003	0.795	7
CTA	0.133	0.133	*	-	-	-
CPA	-	-	-	-	-	-

The SemTab 2021 results of our team

- **Result for CEA**

	F1	Precision
Without Repair	0.502	0.502
Repair	0.692	0.692

After adding noise mentions repair mechanism, we can see that both F1 and precision are greatly improved, which proves the effectiveness of GBMTab.

- **Result for CTA**

The introduction of encoding models and elements in primary key column appears to regularize candidate list at the semantic level and give less weight to the coarse-grained candidates.

Conclusion

- The iterative probability propagation graph model has obvious effects in entity disambiguation. The candidate generation as its upstream task has a greater impact on the disambiguation result. Spelling correction and noise detection in the CEA task can improve the performance of the CTA task.
- The size of table has a great influence on processing speed.
- The application of BERT embedding and property intersection helps to improve the results of CTA task.

References:

[1]. http: //webdatacommons.org/webtables/