3 SYSTEM OVERVIEW

Figure 1 gives an overview of the system architecture of Con2KG, which consists of two main modules:

3.1 Knowledge Mining Module

We divided this module into three phases:

- **Pre-Processing and Entity Extraction**: In this component, we first pre-process the noisy and unstructured data using NLP techniques. We employ sentence simplification module for complex sentences and part-of-speech approaches using Stanford Core NLP framework to revive missing phrases. Additionally, we use dependency parsing tree for rebuilding the syntactic and semantic structure [7]. To deal with abbreviations, we exploit rule-based heuristics and utilize a proprietary vocabulary list. Secondly, we discover the entities such as the type of company, recruiter type, important dates extraction, type of job using a hybrid combination of Stanford NER, libraries, dependency parser, and pattern-based heuristics.
- **Context mapping**: This component aims to describe the contextualization and polarities of the entities extracted from the previous steps. For Example, *"Candidate should be a Post-Graduate. Freshers cannot apply"*. We apply the entity extraction algorithm and remove 'Freshers'(negative polarity) from the list of entities extracted (Post-Graduate, Freshers). We also enhance our module by incorporating contextual information such as preferred candidates (Experienced, Fresher), etc. using Dependency Parser [7].
- **Triple Extraction and Ontology Constitution** : In this module, Firstly, we perform triple extraction using state-of-the-art OpenIE5 [8]. It extracts triples (subject, predicate, object) in a sentence by using pattern templates. In this, we identify the relations and its associated arguments in a sentence without using either prior domain knowledge. For Example, a sentence is *"Candidate should have experience of 4 years"*. After triple extraction we get *("Candidate", "should have experience of", "4 years")*. To add on missing entities, we also defined static relationships for the concepts whose triples are not extracted using OpenIE5. Secondly, we employ a hierarchical structure to these extracted key concepts and hidden nuances. We manually curated concepts related to companies, qualifications, etc. from the structured fields into the Ontology. We use the Louvain algorithm [1] to detect communities with the highest corresponding modularity and iteratively split these communities if the new partition had a positive modularity value. After ontology construction, we apply the state-of-the-art clustering approaches [4] to these entities capturing semantic information.

3.2 Knowledge Integration Module

In this module, we represent all of the extracted knowledge and store it into efficient graphical storage. All the entities (subjects and objects) are nodes and relationships are edges in the graph. Con2KG discovers and easily traverse through millions of nodes and edges using Neo4j Cypher Query Framework [10].

4 EVALUATION

We randomly selected 310 jobs from our legacy dataset containing 4719 sentences to evaluate the quality and quantity of the triples extracted. Based on the results, Con2KG can extract 1.72 triples per sentence on an average. We assess these triples and found 82% precision, 68.23% recall, and F-measure of 74.46%. We also analyze that triple extraction causes 0.05% errors due to incomplete triples and 0.20% due to no triple extraction for most of the sentences. Based on the preliminary analysis, errors in triple extraction occurs due to complexity in unstructured text and relations which are not identified clearly. Apart from these challenges, we still achieve approximately 74% both in terms of quantity and quality. Note that this evaluation methodology is also followed by T2KG [5].

5 FUTURE DIRECTIONS

Con2KG can exploit the entity and its relationships to predict and personalize the job suggestions in the recruitment domain using Entity Cards [9]. We can also query Con2KG and the complex data at the real-time to provide smart knowledge and detect false facts.

ACKNOWLEDGMENTS

This work is supported in part by SERB, FICII and InfoEdge India Limited. We would like to thank the members of the Analytics team (InfoEdge) and Precog, who gave us continued support throughout the project.

REFERENCES

[1] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, 10 (2008), P10008.

[2] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD '08)*. ACM, New York, NY, USA, 1247–1250. https://doi.org/10.1145/1376616.1376746

[3] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, Jr., and Tom M. Mitchell. 2010. Toward an Architecture for Never-ending Language Learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI'10)*. AAAI Press, Palo Alto, California, USA, 1306–1313. http://dl.acm.org/citation.cfm?id=2898607.2898816

[4] Stephen C Johnson. 1967. Hierarchical clustering schemes. *Psychometrika* 32, 3 (1967), 241–254.

[5] Natthawut Kertkeidkachorn and Ryutaro Ichise. 2017. T2KG: An end-to-end system for creating knowledge graph from unstructured text. In *Workshops at the Thirty-First AAAI Conference on Artificial Intelligence*. AAAI Press, Palo Alto, California, USA, 743–749.

[6] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web* 6, 2 (2015), 167–195.

[7] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*. Association for Computational Linguistics, Baltimore, Maryland, 55–60.

[8] Mausam Mausam. 2016. Open Information Extraction Systems and Downstream Applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16)*. AAAI Press, Palo Alto, California, USA, 4074–4077. http://dl.acm.org/citation.cfm?id=3061053.3061220

[9] Afroza Sultana, Quazi Mainul Hasan, Ashis Kumer Biswas, Soumyava Das, Habibur Rahman, Chris Ding, and Chengkai Li. 2012. Infobox Suggestion for Wikipedia Entities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, New York, NY, USA, 2307–2310. https://doi.org/10.1145/2396761.2398627

[10] Jim Webber and Ian Robinson. 2018. *A Programmatic Introduction to Neo4J* (1st ed.). Addison-Wesley Professional, Boston, USA.