A Proposed Model for Improving the Performance of Knowledge Bases in Real-World Applications by Extracting Semantic Information

Abdelrahman Elsharif Karrar
College of Computer Science and Engineering
Taibah University, Medina, Saudi Arabia

Abstract—Knowledge Bases are information resources that convert factual knowledge to machine-readable formats to allow users to extract their desired data from multiple sources. The objective of knowledge base population frameworks is to extend KBs with semantic information to solve fundamental artificial intelligence problems such as understanding human knowledge. Information extraction entails the discovery of critical knowledge facts from unstructured text, which is important in the population of knowledge bases. The objective of this paper is to explore the concept of information extraction as a technique for accelerating the performance of knowledge bases with minimal annotation efforts for real-world applications such as content recommendation during a web search. This entails performing slot filling operations for data collection from large KBs and applying probabilistic estimations to determine the accuracy of the new information. The results are then used to explore the feasibility of applying knowledge bases to real-world tasks such as user-centric information access by encoding entities with deep semantic knowledge.

Keywords—Semantic information extraction; knowledge base; slot filling; content recommendation

I. INTRODUCTION

Knowledge Base (KB) refers to a specially designed resource for gathering and processing knowledge in logical statement formats that define the relationship between graphical entities. Knowledge Bases utilize a relational knowledge representation framework implemented on artificial intelligence, logic, and semantic networks [1]. Facts representation through KBs follows the guidelines by Resource Description Framework (RDF) in the definition of variable relationships among entities, predicates, and values forming triples such that entities represent people or objects, predicates define entity relationship, and values represent other entities, types, attributes, and values [2]. Triples represent existing facts as illustrated in Table I.

Triples in a knowledge base can be aggregated into a graph composed of directed edges representing relationships and nodes representing values and entities. Edge directions reflect the subject entities in specific triples in the condition of two entities. This implies that edges bridge subject entity to object entity. Different edge types are used to represent various relations through structures known as Knowledge graphs, which enhance the visualization and comprehension of KG structures. [3]

DBpedia is an example of a Knowledge Database, which has been developed by research communities to provide an effective framework for knowledge representation as shown in Fig. 1 [4] [5].

Technology companies such as Google, Microsoft, Facebook, and Yahoo construct and manage in-house Knowledge Bases to perform functions such as answering questions and data querying. The most common knowledge bases operated by technology companies include the Microsoft Graph Satori, Facebook Entity Graph, and Yahoo Knowledge graph illustrated in Table II alongside their relation types, number of entities, and the volume of facts [6].

<table>
<thead>
<tr>
<th>Entity</th>
<th>Predicate</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Donald Trump</td>
<td>Age</td>
<td>75</td>
</tr>
<tr>
<td>Donald Trump</td>
<td>Profession</td>
<td>Politician, Actor</td>
</tr>
<tr>
<td>Donald Trump</td>
<td>Starred in</td>
<td>Apprentice TV show</td>
</tr>
<tr>
<td>Apprentice</td>
<td>Genre</td>
<td>Reality Competition</td>
</tr>
<tr>
<td>Apprentice</td>
<td>Release Date</td>
<td>January 2004</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Knowledge Base</th>
<th>Entities</th>
<th>Relation Types</th>
<th>Facts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Knowledge Graph</td>
<td>570 million</td>
<td>35,000</td>
<td>18 billion</td>
</tr>
<tr>
<td>Yahoo Knowledge Graph</td>
<td>3.4 million</td>
<td>800</td>
<td>1.391 billion</td>
</tr>
<tr>
<td>Freebase</td>
<td>40 million</td>
<td>35,000</td>
<td>637 million</td>
</tr>
<tr>
<td>DBPedia</td>
<td>4.6 million</td>
<td>1,367</td>
<td>539 million</td>
</tr>
<tr>
<td>YAGO2</td>
<td>9.8 million</td>
<td>114</td>
<td>447 million</td>
</tr>
<tr>
<td>Wikidata</td>
<td>18 million</td>
<td>1,632</td>
<td>66 million</td>
</tr>
</tbody>
</table>
Knowledge bases differ from traditional databases in their approach to information management since they are focused on “tables” and “records”, which make them efficient when the discovery of new information is not a priority [7]. Knowledge bases are particularly important in domains where the flexibility to link multiple types of information is required. Some of the unique advantages of knowledge bases over traditional data warehouses include:

- **Entity-centric**: All data is stored based on entity relevance.
- **Schema-less**: There are no prior requirements for a schema in the knowledge structure.
- **Metadata Rich**: This contains self-describing metadata streams, which can be easily scaled and integrated across multiple domains.

### A. Applications of Knowledge Bases

Knowledge Bases allow for the semantic structuring of computer-readable information, which is a valuable requirement in the construction of intelligent systems [8]. Knowledge bases are a source of power to various big data applications in multiple scientific and commercial domains such as the integration into Google search engine, which stores approximately 0.57 billion entities and 18 billion facts [9]. The Google Knowledge Graph plays an important role in the identification and disambiguation of textual entities to generate enriched search results by semantic structuring of summaries while providing links to related content during explanatory search [10]. Companies typically rely on knowledge bases in gathering information about various entities and their relationships for optimal reuse efficiency in a domain. Knowledge bases are typically used in querying and displaying entity information, recognizing and extracting context, linking entities to data sources and content, discovering and suggesting related information, semantic parsing, and answering questions in technology platforms such as social media and AI-driven virtual assistants.

The role of knowledge bases in utilizing semantic information generated from knowledge graphs to enrich search results is an important milestone towards the transformation of text-based search engines such as Google into semantically-aware question answering platforms. The concept of knowledge graphs has been prominently demonstrated in Watson; a question-answering platform developed by IBM. Watson used a combination of information sources including Freebase, DBpedia, and YAGO to win the game of Jeopardy against a team of human experts [11]. Structured knowledge repositories are integrated into digital assistants such as Amazon Echo, MS Cortana, and Siri by Apple. Knowledge bases such as Freebase store general data generated by its community members from multiple sources including wiki contributions. Knowledge bases have been applied in the Internet Movie Database (IMDb), which is an online storage platform for information related to video games, television programs, and films including character biographies, reviews, crew information, and plot summaries [12].

### B. The Concept of Information Extraction

Information Extraction (IE) refers to a process through which structured data is generated from semi-structured or unstructured machine-readable formats [13]. The traditional information extraction systems are used for the efficient
The role of SLU techniques in knowledge bases is to perform slot filling tasks and user intent determination, especially in call routing systems, which are integrated with utterance classification capabilities whereby a speech utterance $S_r$ is categorized into one of $M$ semantic categories, $\hat{C}_r \in \mathcal{C} = \{C_1, \ldots, C_M\}$ given that $r$ represents the utterance index [16]. Researchers have recently developed an advanced slot filling method that involves framing tasks in the form of sequence classification problems to identify the phrase boundaries and labels in a semantic template through deep learning [17] [18]. Slot filling tasks in SLU are defined in the Knowledge Base Population (KBP) whose objective is consolidating information from a large multisource corpus for specific attributes of a query entity. Knowledge graphs are powerful and valuable tools for simplifying research tasks such as computing entity weights to allow the allocation of probabilistic weights in the process of enriching semantic knowledge when detecting SLU relations [19] [20] proposed advanced techniques for processing search queries through semantic parsing in multi-turn dialog systems based on unsupervised natural language processing models.

B. Confidence Estimation in IE Systems

According to [21] confidence estimation refers to a machine learning technique that is used to estimate the confidence scores of a specific output in applications such as machine translation and semi-supervised extraction of relations. The confidence scores of output from speech recognition machines can be computed using a maximum entropy model as described by White and Markov models for singleton tokens.

Another research paper [22] proposed an efficient confidence estimation approach for IE outputs based on machine learning models. This approach worked by computing confidence scores for both multi-field records and extracted fields based on the linear-chain Conditional Random Field (CRF) framework.

However, the machine learning approach is simpler compared to the slot filling technique, which performs complex tasks such as sophisticated inference and coreference resolution across multiple documents [23]. Inaccurate values extracted in the slot filling operations for KBPs in multiple systems are filtered using techniques such as weighted voting, unsupervised multidimensional truth-finding, heuristic rules, and supervised learning [24].

C. Rich Annotations

Natural Language Processing (NLP) operations such as extracting information can be improved by leveraging user reviews to customize a system to perform personalized searches [25]. Since user reviews may not be readily available, labels created by human annotators, which apply to a range of supervised learning methods can be used to customize the information retrieval system as proposed by [26]. In this case, the traditional machine learning paradigm may be incorporated with a privileged knowledge model to enable the system to accommodate more annotator labels. Recent studies observe an issue with the underutilization of human annotators due to the inclusion of rich annotations into various classification problems [27] [28].

The approach to learning new information through error corrections is conceptualized from the Transformation-based
D. Content Recommendation by user Profiling

The effectiveness of information extraction techniques for improving Knowledge [30] Bases may be improved through user profiling using factoring machines and recommendation systems. Research studies suggest that the primary objective of user profiling in IE systems is to align user interests with the recommended items for example in online shopping platforms such as Amazon, the content recommendation in Netflix, or web search customization for enhanced user experience in Google [31].

The functional mechanism of recommendation systems in data extraction may be through content-based recommendation or collaborative filtering, which utilizes matrix factorization and nearest neighborhood techniques to compute user collaboration scores [32].

However, content-based recommendation algorithms work by extracting the unique and dominant attributes that explicitly link users to items, especially in systems with multiple cold start items [33].

According to [34] and [35] researchers have proposed improved approaches for content recommendation by user profiling based on activity ranking, hypergraph learning, latent factor models, probabilistic models, and spatial-temporal model.

A study by [36] observes that developers are now more focused on embedding recommendation and user profiling systems with hierarchical knowledge repositories for the creation of personalized entity recommendations based on knowledge and user activity log obtained from freebase.

A content-based recommendation model proposed by [37] implements a spreading activation algorithm on the DBpedia categorization structure to extract [38]information on user preferences and interests. This technique was later applied to music entity recommendation by Linked Data Semantic Distance (LDSD) with DBpedia by [39] and to movie recommendation by [40]. The recommendation systems are capable of modeling user preferences by extracting information from multiple sources such as implicit and explicit profiles. Deep semantic knowledge provides a framework for extracting rich contextual knowledge of user queries by analyzing the data networks to identify entities in which the users are interested.

The information extraction framework proposed in this paper is consistent with a study [35] which focused on modeling user preferences for customized content recommendation in large knowledge bases primarily relying on data from the Yahoo Knowledge Graph.
The training dataset for the framework used in this work is created by searching the internet for related entity pairs in a knowledge graph using the model proposed by [43]. Assuming a web search returns AS as the set containing entity pair a and b, SAS, a subset of AS having

\[ SAS = \{ s : s \in AS \land (s, a) \land (a, b) \} \]

where \( \land \) is true when \( n \) is a substring of string \( m \). The sentences are then post-processed for the augmentation of relation tags from the knowledge base because some instances may contain multiple relations. For example, if two relations; place of birth (New York, USA) and date of birth (October 17, 1983) about “Brad Hudson” is extracted, post-processing would produce the following instances complete with tags instead of tag-less instances: Brad Hudson was born on \(<\text{date_of_birth}>October 17, 1983, </\text{date_of_birth}> in \text{place_of_birth}>New York, USA<\text{place_of_birth}>.

B. Classifying the Personal Assertions

In this experiment, 10 million utterances are extracted from Microsoft KBs and query logs. Factual relations are then mined by extracting personal assertions containing factual relations through the following in the following pattern; ‘I am a *, I have a * I live * I was born * I work*’. A random subset of the extracted is selected and annotated whether it satisfies the requirements; it is a personal assertion, invokes relations, and entities can be extracted from the invoked relations. The final dataset contains 12,989 personal assertions out of which only 1,811 utterances contain one or more pre-defined relations. A 10-fold cross-validation technique is then used to create 10 random subsamples whereby 9 subsamples are set aside for training and 1 subsample is retained as a validation set. Cross-validation operations are performed only once on each subsample. From the 236,724 collected samples, 234, 650 are classified accurately (99.12%) and 2,074 are classified inaccurately. This implies that SVM is an efficient classifier for personal assertions.

C. Detecting Relations

The performance of relation detection functionality is determined by testing the models trained using the annotated datasets extracted in the previous section in two scenarios; supervised baseline and unsupervised baseline. A precision model P@N was used in the evaluation given that \( N \) represents positive relations in a given set. From the supervised baseline where 2-fold cross-validation is utilized and the model trained on randomly assigned utterances to two data sets, 84.32% \( P@N \) upper bound precision is obtained while the unsupervised technique attains 42.85% \( P@N \) upper bound precision.

D. Slot Filling

The supervised technique was used to perform the slot filling operation due to the variations in semantic annotation mechanisms of the sampled set. The slot F-Measure model was applied to the CoNLL processing script, which attained 68.34% performance efficiency. The model achieves higher performance efficiency when applied to minimal annotations and nontrivial tasks as illustrated in Table III.

### Table III. Performance Efficiency Results for Slot Filling and Relation Detection in Data Extraction

<table>
<thead>
<tr>
<th>Relation Type</th>
<th>Count</th>
<th>Relation Detection</th>
<th>Slot Filling</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Unsupervised</td>
<td>Supervised</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Precision@Count (%)</td>
<td>Precision@Count (%)</td>
</tr>
<tr>
<td>place_of_birth</td>
<td>8</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>religion</td>
<td>8</td>
<td>0.00</td>
<td>50.00</td>
</tr>
<tr>
<td>ethnicity</td>
<td>17</td>
<td>0.00</td>
<td>70.59</td>
</tr>
<tr>
<td>employment_history</td>
<td>40</td>
<td>7.50</td>
<td>52.50</td>
</tr>
<tr>
<td>nationality</td>
<td>47</td>
<td>0.00</td>
<td>63.83</td>
</tr>
<tr>
<td>profession</td>
<td>61</td>
<td>0.00</td>
<td>54.10</td>
</tr>
<tr>
<td>gender</td>
<td>63</td>
<td>6.35</td>
<td>82.54</td>
</tr>
<tr>
<td>date_of_birth</td>
<td>73</td>
<td>46.58</td>
<td>75.34</td>
</tr>
<tr>
<td>places_lived</td>
<td>121</td>
<td>2.48</td>
<td>68.59</td>
</tr>
<tr>
<td>sibling_s</td>
<td>248</td>
<td>86.29</td>
<td>90.32</td>
</tr>
<tr>
<td>children</td>
<td>260</td>
<td>23.08</td>
<td>87.31</td>
</tr>
<tr>
<td>parents</td>
<td>401</td>
<td>19.95</td>
<td>86.78</td>
</tr>
<tr>
<td>spouse_s</td>
<td>464</td>
<td>82.11</td>
<td>94.39</td>
</tr>
<tr>
<td>Total</td>
<td>1811</td>
<td>42.85</td>
<td>84.32</td>
</tr>
</tbody>
</table>
IV. CONTENT RECOMMENDATION BY USER PROFILING

The evolution of the Web has positioned the internet as a crucial player in providing users with access to information from multiple sources. Information overload is one of the greatest challenges of the web hence the need for content recommendation to match user interests. Despite the monumental milestones in the design of recommendation systems, there are significant challenges in availing users of high-quality information. This section explores a high-performance content recommendation model for efficient information extraction from knowledge bases. The core objective of user modeling in this framework is to understand their current preferences and predict future interests in contextual applications such as sports databases. The data used in this experiment is obtained from Yahoo News Streams, which contain information such as the sequence of websites that a user has visited as expressed in the (1) for a typical user $u$;

$$L^u = \langle w^u, w^u, \ldots, w^u, \ldots, w^u \rangle$$  \hspace{1cm} (1)

Such that $w^u$ represents the websites visited by user $u$ at time $t$.

Unstructured information containing attributes such as the user location, language, identity, demographics, timestamps, and click/skip labels. Additionally, the Wikipedia Knowledge Graph was used as a knowledge source for enriching feature space by monitoring evolving sources and wrapping different sources.

A. Modeling user Profiles

A high-level Pipeline algorithm is utilized to model user interests and predict preferences and FastEL software is used for linking entities. A separate entity augmentation algorithm is used to extract entities from user logs then link them to the entities in the Wiki KB. The following code is executed to perform this operation;

Input: A sample user opened document $D$ stored in a Global KG $G$, which contains relation triples defined by $\sigma = E_u, \rho, E_b$ such that $\rho$ represents a relation predicate for $n$ iterations $m$ maximum augmented entities.

1: Generate initial entities $E = \{e\}$ from $D$
2: repeat
3: Augment entities using facts from $G$
4: Re-score interest weights of augmented entities
5: until converged or reach $n$ iterations
6: return top $m$ augmented entities from the list

Named entities can be extracted from the visited web pages and linked to related Wiki entities based on the user logs. However, the entities may not provide adequate information on user interests hence cannot accurately predict future preferences hence the need to leverage the Yahoo Knowledge Graph to augment the entities into relational facts with a higher degree of accuracy. Once the entities are augmented and retrieved, a decayed interest weight is then assigned to indicate the lowest probability that user interests lie in a particular category.

B. The Framework for Profiling users

According to [44] the user profiling model used for content recommendation in search engines utilizes Factorization Machines (FM) to perform latent factoring and matrix factorization in recommender systems. A latent space for every user is constructed to allow for the differentiation of user preferences in the process of learning from the unstructured dataset. A factorization-machine-based latent factor framework is used to decompose every user profile shared and personalized latent factors. The process of mapping profiles into latent factors is standardized for every user hence making it possible to enrich the information for those with minimal interaction data.

C. Experiment

The experiments are based on a sample of 32.09 billion user logs collected from Yahoo News Stream over one month. The user profiles are evaluated for quality by splitting the dataset into training and testing groups based on event timestamps. Data sets from the first three weeks (23.68 billion events) are used for training while data from the fourth week (8.42 billion events) is used for model testing. For the training dataset, each user profile is ranked and performance evaluated based on ground truth labels, which may be positive or negative.

Inner product values are used between item features and user profiles to generate the ranking scores of each user-item pair. The items are then ranked as positive if they have a higher ranking otherwise negative based on metrics such as the Area under the Curve (AUC), Mean Reciprocal Rank (MRR), and Mean Average Precision (MAP) as defined in the (2), (3) and (4);

$$MRR = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{r_i}$$  \hspace{1cm} (3)

$$MAP = \frac{1}{m} \sum_{i=1}^{m} \frac{\sum_{k=1}^{P(i)} P(k)}{P(i) N_i}$$  \hspace{1cm} (4)

Given that $P(k)$ represents precision at $k$, $n_i$: user-related links, $u_j r^k$: ranking of the links that were clicked first by user $u_i$, $P_i$: the clicked links, and $N_i$: non-clicked links in the profile for user $u_i$.

When the number of iterations is adjusted to 1, it achieves about 193% relative and 10% absolute performance improvement in mean average precision; 191% relative and 17% absolute performance improvement in mean reciprocal rank, which is significantly high compared to the baseline system, which obtained 12% relative and 7% absolute performance improvement. Mean average precision computes the average precision scores for listed items while the mean reciprocal rank calculates the inverse position of the initially ranked relevant items. Therefore, both MRR and MAP compute ranking scores for listed items. The area under the curve describes the ratio of false positives and true positives when the threshold parameter is varied suggesting that when
entity ranking and coverage are applied to content recommendation through entity augmentation, it extracts additional related entities enriching the feature space significantly according to [45].

V. DISCUSSION AND CONCLUSION

Technological evolution has led to the rapid adoption of online news platforms as a source of information from a wide range of sources across the globe. Due to the high volume of documents on millions of websites, users face many challenges finding their articles of interest or any other precise information. Knowledge Bases such as Wikipedia are rich information resources for users seeking knowledge in various fields including culture, technology, science, and history. This study sought to improve the efficiency of knowledge bases by analyzing the statistical frameworks for building user-centric KBs and extracting personal facts from user utterances through personal assertion classification.

The study also sought to understand how the accuracy of information extracted from knowledge bases can be validated using a maximum entropy framework. Consequently, a framework for rich annotation-guided learning was developed as an approach for improving the efficiency of knowledge basis through information extraction [13]. The annotation framework was designed with a capability for feature enrichment, which allows for the analysis of relative efficacy and scalability of slot filling operations in KBP settings. A review of previously published studies demonstrates that a slight increase in the annotation period improves KB performance significantly. The study also sought to investigate how knowledge bases can be improved to advance tasks such as content recommendation based on the users’ online activity. The experimental findings for these improvement operations in knowledge bases suggest that refining information extraction techniques is an efficient approach to improving the performance of knowledge bases.

VI. FUTURE WORK

While researchers have made significant progress towards the understanding of knowledge base architectures, various gaps need to be filled, especially on the categories of knowledge possessed by human beings. Current literature does not provide detailed representations of facts based on common sense and procedural knowledge. Knowledge representation through reasoning and learning remains an important aspect of future studies on the integration of machine learning and artificial intelligence capabilities to information extraction from knowledge bases. Other relevant fields for future research include the population of personal knowledge graphs, confidence estimation for knowledge bases, and guided learning for rich annotations.

REFERENCES


