submitted version
DOI of published version on Future Generation Computer Systems
https://doi.org/10.1016/j.future.2025.108063

Entity Ablation of Knowledge Graphs: Impact on Information Quality and Sustainability [Experiment, Analysis & Benchmark]

Edoardo Ramalli
Politecnico di Milano
edoardo.ramalli@polimi.it

Carlo Alberto Bono
Politecnico di Milano
carlo.bono@polimi.it

Camilla Sancricca
Politecnico di Milano
camilla.sancricca@polimi.it

Cinzia Cappiello
Politecnico di Milano
cinzia.cappiello@polimi.it

Marco Comuzzi
Ulsan National Institute of Science and Technology
mcomuzzi@unist.ac.kr

Barbara Pernici
Politecnico di Milano
barbara.pernici@polimi.it

Monica Vitali
Politecnico di Milano
monica.vitali@polimi.it

ABSTRACT

Knowledge Graphs (KGs) are powerful tools for organizing and extracting knowledge in various domains. However, as KGs become larger and more complex, they can pose challenges to operations such as query answering and machine learning. These challenges have led to the development of techniques for KG reduction. This paper analyzes the impact of ablation on KG size reduction by systematically evaluating how different entity ablation strategies, combined with various KG embedding techniques, affect the quality of the output of downstream tasks. Specifically, the experimental study focuses on link prediction (LP), a key task for deriving new knowledge from KGs. The study considers the impact of entity ablation by analyzing the output quality in terms of LP accuracy and information loss in the KG, and sustainability gains, such as reduced energy consumption, achieved through size reduction. The results demonstrate that selective and controlled ablation can preserve the semantic integrity of a KG while maintaining output quality and improving sustainability. In addition, based on the experimental study, we present a machine learning-based approach to evaluate the impact on the sustainability of the different design choices, considering the characteristics of the KG and the desired quality of the results.

PVLDB Reference Format:
Edoardo Ramalli, Carlo Alberto Bono, Camilla Sancricca, Cinzia Cappiello, Marco Comuzzi, Barbara Pernici, and Monica Vitali. Entity Ablation of Knowledge Graphs: Impact on Information Quality and Sustainability [Experiment, Analysis & Benchmark] . PVLDB, xx(x): XXX-XXX, 2025.
doi:XX.XX/XXX.XX

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/edoardoramalli/KGE_Quality.

1 INTRODUCTION

Knowledge Graphs (KGs) are increasingly developed and used for various tasks in several domains. They allow for answering complex queries and inferring available data. The size of recent KGs and their constant expansion introduces crucial concerns. They regard the computational efficiency of tasks performed using them and the environmental impact and cost of such tasks. For instance, a single supervised learning training on a medium-sized KG (100,000 entities) can take several hours, even with cutting-edge hardware [1]. Moreover, modern KGs contain millions of entities and tens of millions of links [2], also requiring a linear space complexity to store and train machine learning (ML) models [3].

Researchers in different fields have advocated the need for methods to reduce KG size to facilitate exploration, analysis, and knowledge extraction tasks, while preserving the information content [4–7]. Reducing a KG is not straightforward, as removing certain parts can alter the information it conveys. A critical question is how to trade off the smaller *complexity* of the reduced graph with the *information loss* due to the reduction while maintaining the KG's semantic learning potential and minimizing computational costs.

KG reduction can be achieved through information aggregation (summarization) or selection (e.g., node ablation) [8]. In KG summarization, entities are aggregated into more general ones according to some rule, resulting in a more compact KG. Since the entities of an aggregated KG are not the same as the original ones, KG summarization precludes some typical KG analysis tasks, such as query answering or link prediction. Alternatively, selection involves obtaining a new KG by removing entities, i.e., *node ablation.*

This paper focuses on link prediction (LP) and designs and discusses experiments to identify an effective way to perform node ablation in KGs to analyze the trade-off between the link prediction performance and the KG information loss. LP task requires that a KG is encoded using some Knowledge Graph Embedding (KGE) model. KGE models aim to learn low-dimensional vector representations of entities and relations in a KG, enabling the capability to learn the semantics of the data for tasks such as LP, entity recommendation, and question answering.

In other words, this work aims to identify controlled KG node ablation strategies that preserve the quality of the task results without requiring prior domain knowledge of the KG by measuring the impact on the LP results and information loss.

The design of such an experiment entails several design dimensions and several choices to be made to keep the size and the scope of the experiment feasible. First, the LP leverages the embeddings of a KG. The efficiency and precision of these embeddings are crucial for the effectiveness of downstream applications. In the experiment,

Proceedings of the VLDB Endowment, Vol. xx, No. x ISSN 2150-8097.
doi:XX.XX/XXX.XX

we consider multiple KGE techniques and evaluate their performance under different settings. We investigate different ways of selecting which nodes to remove from a KG. We consider node ablation based on geometric and spectral centrality measures. For each type of measure, we study the effects of ablating either the nodes with higher or lower centrality against a baseline in which nodes are randomly removed. As far as datasets are concerned, our experiments are conducted on widely used KG datasets, using standard evaluation metrics for link prediction.

The results suggest that by ablating the less central nodes of a KG, it is possible to obtain performance in the LP task close to that obtained with the original KG while limiting the loss of information in the KG and thus maintaining an adequate semantic representation. Based on these results, we propose a machine learning approach to suggest the optimal centrality metric for node ablation and determine the appropriate ratio of less significant nodes to delete while satisfying specific information loss constraints. This approach also estimates sustainability in terms of energy and cost savings associated with the suggested KG reduction.

The paper is structured as follows. Section 2 discusses the state of the art on graph reduction and link prediction. Section 3 presents the methodology used to design the experiments; the experimental setting is described in Section 4. The results are illustrated and discussed in Section 5. Based on these results, Section 6 presents a machine learning-based approach to evaluate the design choices based on the characteristics of the data and the desired quality of the results. Conclusions are drawn in Section 7.

2 BACKGROUND AND RELATED WORK

Motivated by the complex and heterogeneous structure of KGs, research on automated reduction techniques for efficient entity description has emerged recently [9, 10].

There are two ways of reducing a KG: aggregating or simplifying, with the first option being widely studied in literature [10]. The main idea of graph aggregation – or summarization – is to generate a succinct and meaningful description of a given KG to reflect its main content while retaining the information [11]. Research in this area focuses mainly on scalability, performance (e.g., query time), and reconstruction [6, 12]. When aggregating a KG, its entities are condensed into more general ones based on some characteristics, resulting in a more compact KG.[1] Graph summarization has also been addressed in a framework of inductive rules derivation, describing what is "normal" in a KG, which can be used to best compress the KG itself [13]. One of the main limitations of the presented methods is that they usually give little control over which of the original entities are retained, if any. Moreover, the absence of the original entities from the summarized graph implies that it cannot be utilized for the LP task.

Another approach involves simplification methods, which maintain a subset of the original KG by removing nodes or edges, thus resulting in a sparsified graph (*ablated graph*). In this case, an ML task (e.g., LP) can be performed on the ablated graph [10]. Simplification methods have leveraged structural properties of the graph to reduce it [8], with techniques based on node or edge degree [14], edge resistance [15] or spectral sparsification [16]. *Node ablation* is defined as a systematic removal of KG entities based on some measure of interest, such as *Centrality measures* [17], that are structural, general-purpose, and highly scalable measures to estimate the importance of nodes in a graph.

The concept of *information loss* is generically defined as "*the amount of 'useful information' represented by the graph for the task at hand*" [18]. Such loss can be calculated by considering the edges lost or spuriously created in the reconstruction, weighting them by their centrality [19]. Methods to account for user preferences in the graph summarization process have also been proposed, typically by minimizing *information loss*. This loss is measured by evaluating the structural similarity between the original graph KG and a graph KG' reconstructed from the summarized graph [20]. Algorithms for selecting representative nodes by minimizing the graph reconstruction error have also been proposed [21].

This work aims to observe the impact of ablation on the LP task. LP is a KG completion task to enhance a KG's completeness and, hence, its practical utilization. LP leverages existing relationships to infer new ones in a KG. This is motivated by the observation that KGs, which range from small, curated datasets to low-quality crowdsourced data, can have highly variable quality [22]. Several research studies have focused on quality issues in KG, such as inaccuracies, inconsistencies, and validity, together with associated data quality dimensions and metrics [23, 24]. Empirical studies have also been proposed to assess the correctness and completeness of KG data using LP [25]. Automated KG refinement using machine learning approaches has been an active research topic in recent years [26]. The LP performance is adopted in the literature to foreshadow how effectively a KGE captures the semantics of a KG [27]. However, LP performance alone does not allow for identifying which portions of the KG contribute most significantly to achieving a given task performance level. Following this reasoning, systematically isolating a "representative" part of the KG through node ablation can potentially balance performance and efficiency when training KGEs, while also shedding light on the characteristics of the learning process.

LP implies the use of KGE models. For over a decade, methods for embedding the symbolic representations of knowledge bases into continuous vectors have been continuously developed [28]. Knowledge Graph Embedding (KGE) can support a variety of tasks, such as information retrieval, prediction, and discovery. The ability of KGE models to learn low-rank representations of relationships and entities has sparked research on different models and model families to perform this task effectively. Research also explores the influence of graph structural features on the learning and predictive capabilities of KGE models [3, 29], occasionally exploring this through context-dependent ablation studies [30]. Comparative analyses are frequently conducted, including surveys of existing approaches [31–33] and comprehensive benchmarks [1, 3, 34, 35]. These efforts are driven by the diversity of available models, hyperparameters, datasets, evaluation criteria, and the challenges associated with replicating the results. The computational and space complexity and the carbon footprint are also evaluated [1]. Alongside, comprehensive software frameworks to train and evaluate KGE models have emerged [36].

[1] As an example, *Rome*, *Florence New York City* could be represented by a *CITY* entity in the aggregated KG.

KGE models differ in representation space, scoring function, encoding models, and other information [3, 31]. Geometric KGE models interpret KG facts as geometric transformations in the embedding space. These models commonly use the Euclidean space to learn a vector representation of entities and relations. The typical model TransE [37] and the more recent RotatE [38] belong to this family of KGE models. Tensor decomposition models, such as ComplEx [39], decompose the KG adjacency matrix in low-dimensional vectors used as embeddings for entities and relations.

To the best of our knowledge, little attention has been paid to balancing the performance and the efficiency of KGE models by training them on simplified versions of a KG. This work investigates this research gap by providing a quality-aware ablation study in the context of KGE, to exploit the trade-off between model expressivity (i.e., quality in terms of task performance and information loss) and training efficiency (i.e., sustainability in terms of energy and cost), without requiring specific domain knowledge for the data [30]. We address these issues by focusing on a systematic evaluation of the trade-offs in applying different KGE techniques and different node ablation ratios based on selected centrality measures, as described in the next section.

3 METHODOLOGY

A *Knowledge Graph (KG)* is a collection $G = \{E, R, F\}$, where E, R, and F are sets of entities (nodes), relationships (edge types), and facts (edges), respectively. A *fact* or *statement*, $f \in F \subseteq E \times R \times E$ is a triple $\langle h, r, t \rangle$ where $h \in E$ stands for the "head", $r \in R$ stands for the "relationship", and $t \in E$ stands for the "tail". Head, relationship, and tail are interchangeably denoted as subject, predicate, and object. A KG is then a collection of facts, where each fact connects two entities through a semantic relationship.

A *Knowledge Graph Embedding (KGE)* aims to represent the relationships and entities in KGs in a low-dimensional and continuous space. It is computed through an ML model that encodes each entity and relationship of a knowledge graph $\mathcal{G}$ into a vector of embedding dimension d, while preserving their semantic meaning. Entities h, t and relationships r are transformed respectively into their embeddings $\mathbf{e}_h$, $\mathbf{e}_t$, and $\mathbf{e}_r$.

A KGE is defined by a *representation space* where entities and relationships are encoded via a *model* that describes how their embeddings interact [3, 31, 40], and a *scoring function* $\mathcal{F}(e_h, e_r, e_t)$ that measures the plausibility of a fact. During training, these components enable the model to learn embedded representations of KG entities and relationships: similar entities should be encoded to similar embeddings.

Training a KGE on a smaller KG reduces computational effort since the number of facts to be learned affects the training time. In this work, KG entities are removed (i.e., *ablated*) to evaluate the impact of the removed information on KGE. One way to assess this impact is to measure the KGE performance on the link prediction task.

Given a set of *observed* triples F in a KG, the *Link Prediction (LP)* task aims to predict valid but *unobserved* triples among all possible existing facts $E \times R \times E$. The LP procedure works as follows: given the embedded representation of a KG, i.e., the KGE, and the scoring function $\mathcal{F}$ of the employed KGE model, LP can infer the missing element of a triple when the other two elements are provided. For instance, if the tail of the triple is missing $\langle h, r, ? \rangle$, the embeddings e_i, with $i \in E$, of all the existing entities in the KG are retrieved. The KGE model's scoring function is then used to calculate a score for each possible combination of the given head and relationship with every potential tail entity $\mathcal{F}(e_h, e_r, e_i)$. These triples are ranked based on their scores, with the lowest score indicating the highest plausibility of a candidate to complete the triple. With the ranking, it is possible to quantify the LP performance.

There are two ways to determine which triples are ablated from a KG: selecting entities or selecting relationships. In either case, all triples containing the selected entities or relationships are removed. For this experimental work, we decide to perform the ablation by selecting KG entities. In this way, we maintain all the relationship types, even if not necessarily with the original relative frequency. Nodes in a graph, which are topologically equivalent to entities, have well-studied structural properties that can be leveraged to identify candidates for removal without requiring specific domain knowledge. Among these properties, *centrality* measures are often used to estimate the importance of an entity.

3.1 Methodological Steps

Experiments have been conducted by following a rigorous methodology composed of the following steps (see Figure 1):

(i) A given KG is randomly split into training and test sets (the elements of the test set are indicated in red in Figure 1.i).
(ii) Centrality measures are computed for each entity in the training set. To exemplify, in Figure 1.ii, the degree of each node is indicated. The centrality measures adopted in this work are discussed in Section 3.3.
(iii) A set of entities is removed from the KG, based on their centrality (ablation step, obtaining an ablated KG, shown in Figure 1.iii). The figure shows the ablated portions in dashed lines for both the training set (in black) and the test set (in red). Different ablation ratios are applied in the experiments, resulting in different ablated KGs (see Section 4).
(iv) The ablated KG is used as a training set for a KGE model (Figure 1.iv). Note that different KGE embedding techniques are evaluated (see Section 3.4).
(v) For each KGE resulting from an ablated KG, graph properties and prediction capabilities are measured and compared with the results obtained using the training set obtained from the original KG (Figure 1.v). Details are provided in (Section 3.5).

The proposed approach for each step is discussed in detail in the following subsections.

3.2 Dataset Split

To generate *unobserved* facts for evaluation, a random split is performed on the considered KGs. This is equivalent to postulating a closed world assumption (CWA) for the original KG and a controlled open world assumption (OWA) for the training set, where unobserved facts are allocated to the test set, but no other fact is unobserved.

To perform the LP on a fact triple $f : \langle h, r, t \rangle$ contained in the original KG and selected for testing, the embedding of h and t should be available. This implies that even if a fact f belongs to the

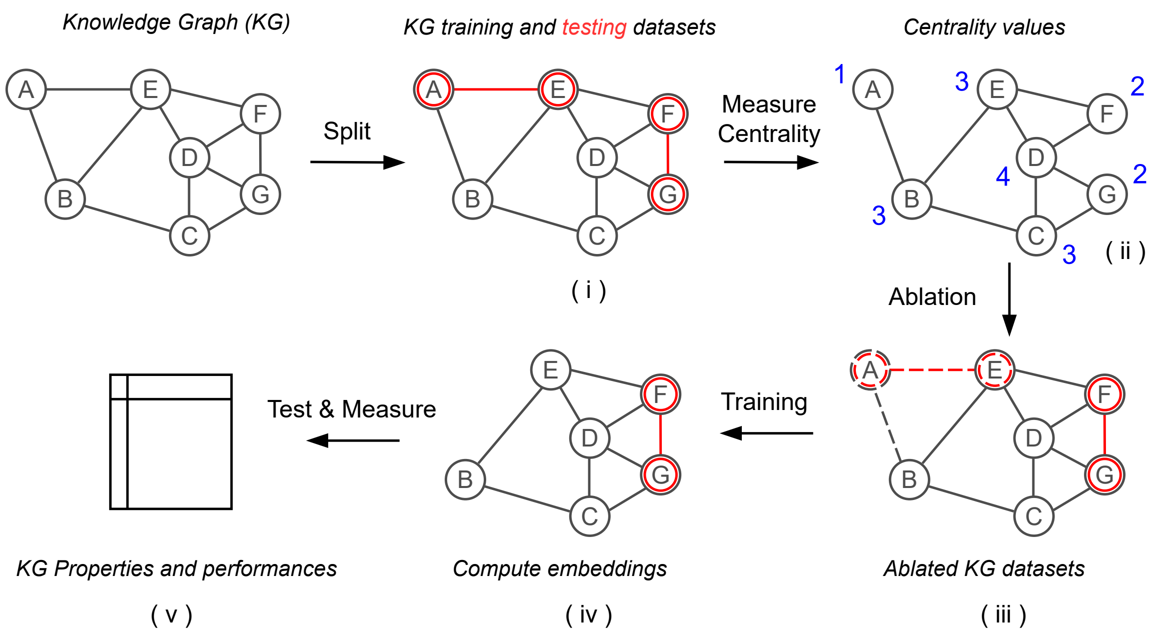

Figure 1: Steps of the methodology

test set only, the entities h and t must also appear in the training set. To guarantee this requirement, the splitting is performed in the following way: first, training triples are greedily selected from the original KG such that each entity is covered in both sets; then, the remaining triples are split randomly, according to the configured split ratio, and assigned to the respective sets. Multiple data splits can be conducted to obtain more reliable performance measures, with the results averaged across the splits.

3.3 Centrality Measures and Ablation

Centrality measures are derived from network analysis to quantify the "importance" of a node in a graph. A multiplicity of measures and types of measures correspond to different notions of centrality. In this experimental analysis, we selected the following centrality measures, which are among the most commonly used in centrality studies [17], computing them on the KG, disregarding the relation type. We then utilize the centrality measures independently in separate experiments to progressively remove quantiles of the nodes/entities based on their centralities.

Harmonic Centrality. The harmonic centrality of an entity $e \in E$ is a geometric measure calculated as the sum of the reciprocal of the shortest path distances from all other entities connected to e.

$$HC(e) = \sum_{\substack{d(e,e_i)<\infty \\ e\neq e_i}} \frac{1}{d(e,e_i)}$$

where $d(e, e_i)$ is the shortest-path distance between the entity e and a connected entity e_i [41].

PageRank. The PageRank centrality (PR) is a spectral measure representing the likelihood of landing on an entity by randomly surfing the structure of a graph. The PR of an entity can be computed iteratively, up to convergence, as:

$$PR(e) := \frac{1-d}{|E|} + d \cdot \sum_{e \in M(e)} \frac{PR(e)}{L(e)}$$

where $M(e)$ is the set of nodes that have an outbound edge to e, $L(e)$ is the number of outbound edges from e, and d is the damping factor, namely the probability that a surfer will keep following the graph instead of jumping to a random node [42].

Random. Although not a centrality measure, we also perform a random ablation to create a baseline to compare it with the other ablation methods.

3.4 Training

The training of the KGE models is performed on every ablated version of the KG obtained with the previous steps. The KGEs are then used in the following step to assess the variation of LP performances and compare the benefit of ablation with semantic information loss.

To obtain representative results, we select a subset of renowned KGE models that belong to the KGE families introduced in Section 2. This work considers TransE [37], ComplEx [39], and RotatE [38] as representative KGE models to learn the embeddings of the KGs. According to [3], these three models balance complexity and performance within the tensor decomposition and geometric family of models.

TransE [37]. TransE models relationships as a translation from head to tail entities in the embeddings space:

$$\mathbf{e}_h + \mathbf{e}_r \approx \mathbf{e}_t$$

This equation is rearranged, and the TransE interaction function is obtained by applying the L_p norm[2]: $f(h,r,t) = -\|\mathbf{e}_h + \mathbf{e}_r - \mathbf{e}_t\|_p$.

ComplEx [39]. Thanks to the formulation of the embeddings in the complex space, ComplEx is capable of learning asymmetric relationships, leveraging the capabilities of the Hermitian product. ComplEx defines its score function as follows.

$f(h,r,t) = Re(\mathbf{e}_h \odot \mathbf{r}_r \odot \mathbf{e}_t)$

RotatE [38]. RotatE models relationships as rotations from head-to-tail entities in complex space:

$\mathbf{e}_t = \mathbf{e}_h \odot \mathbf{r}_r$

where $\mathbf{e}, \mathbf{r} \in \mathbb{C}^d$ and the complex elements of $\mathbf{r}_r$ are restricted to have a modulus of one ($\|\mathbf{r}_r\| = 1$). The interaction model is then defined as:

$f(h,r,t) = -\|\mathbf{e}_h \odot \mathbf{r}_r - \mathbf{e}_t\|$

which allows modeling symmetry, antisymmetry, inversion, and composition.

3.5 Test & Measure

3.5.1 Task Performance Evaluation. In the experiment, we use the LP task as a benchmark to assess the ability of a KGE to infer the validity of a test set of facts Q, which we know to be true by construction (see Section 4.1). The LP task is performed by removing, one at a time, the head and the tail entity for each triple presented in the test set and assessing the capabilities to complete the triples.

To this end, different task performance metrics can be utilized [43, 44]. One commonly utilized metric is *Hits@N*, or *H@N*, which counts how many correct triples appear within the N top-ranked predictions of an embedding model Equation (1). *Hits@N* is calculated as:

$$Hits@N = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\begin{cases}1 & \text{if } \mathrm{rank}_i \le N\\ 0 & \text{otherwise}\end{cases} \in [0,1] \qquad (1)$$

where $\mathrm{rank}_i = \mathrm{rank}(\langle h,r,t\rangle_i)$. The *Hits@N* lies within $[0,1]$ regardless of N; values closer to 1 are better. The literature often uses H@10 as the task performance metric.

3.5.2 Information loss. The ablation of the KG element is not cost-free. The downside of this operation is the loss of some information. This work investigates the impact of removing less central entities from the KG to reduce its size. Therefore, measuring the information loss is fundamental to counterbalance this action.

In the literature, PageRank is commonly used to assess the quantity of information stored in a KG [6]. We measure the information loss (InfLoss) as the sum of the PageRank of the entities removed:

$$InfLoss = 1 - \sum_{i=1}^{K} PR(e_i) \qquad (2)$$

where $E = \{e_1, e_2, .., e_k\}$ is the set of entities remained in the graph after ablation, and PR is the PageRank value of each entity $e \in E$.

[2]The original formulation considers either L_1 or L_2

3.5.3 Sustainability Evaluation. KGE models often demand substantial time and computational resources for training. This is directly related to costs and energy consumption. We expect that reducing the number of nodes of a KG will decrease the computational time and, consequently, cost and energy. To prove this correlation, during our experiments, we also measure the time required and the energy consumed (in kWh) for each training session.

We exploit our results to build ML-based models to recommend, for an unknown KG: (1) the appropriate ratio of entities that can be deleted while keeping a specific information loss threshold and (2) the optimal centrality metric to use to achieve the lowest energy consumption, i.e., the one that saves the largest amount of training time. To define such models, we need to introduce metrics for the Percentage of Ablated Entities and the Percentage of Time Saved (PTS); they are enunciated below:

The Percentage of Ablated Entities (PAE) is computed as $PAE = 1 - \frac{|e_{ablation}|}{|e_{baseline}|}$, where $|e_{ablation}|$ is the number of remaining entities and $|e_{baseline}|$ is the number of entities of the original KG.

The Percentage of Time Saved (PTS) is computed with the formula: $PTS = 1 - TTR$, where TTR is the Training Time Ratio defined as $TTR = \frac{TT_{ablation}}{TT_{baseline}}$, $TT_{ablation}$ is the training time using the ablated KG and $TT_{baseline}$ is the training time of the original KG. This transformation is required to make the results of the experiments obtained with different models and numbers of entities comparable.

4 EXPERIMENTAL SETUP

In this section, we present the experimental setup for the ablation study, illustrating the datasets used, the derivation of the test set, and the implementation of the tests. In our setup, a single experiment is defined as a dataset split, ablated with a specific percentage, and fed to an embedding model. From each dataset-split-ablation-model combination, we extract the task performance, information loss, and training efficiency metrics. This work presents the main results from 1 782 dataset-split-ablation-model combinations, with a total runtime of ~120 days.

4.1 Datasets

We propose a domain-independent approach to entity ablation. Thus, we performed the experiments on three widely used KGs with different characteristics:

- *Freebase (FB)* [45] contains general human knowledge, with more than 100 million assertions and more than 4 000 topics, including people, media, places, and many others. In the experiments, we adopted the FB15k dataset that contains 1 345 relations, 14 951 entities, and 592 213 facts.
- *WordNet (WN)* [46] is a lexical reference system whose design is inspired by psycholinguistic and computational theories of human lexical memory. It contains, for example, English nouns, verbs, adjectives, and adverbs grouped into synonym sets of lexical concepts. We adopted the WN18 dataset with 18 relations scraped from WordNet, 40 943 entities, and 141 442 facts.
- *Yet Another Great Ontology (YAGO)* [47] is an ontology containing more than 1 million entities and 5 million facts. Facts were extracted from Wikipedia and unified with WN

to add knowledge about the entities (e.g., persons, organizations, or products) with their semantic relationships. The present work employs the YAGO310 subset, which contains 37 relations, 123 182 entities, and 1 179 040 triples.

4.2 Experiments Implementation

Split. The original KG is randomly split three times in the training and test set (respectively, 80% and 20% of the triples).

Node ablation. The criteria for selecting the KG entities to be ablated include PageRank, Harmonic Centrality, and random selection as the baseline. The ablation ratios employed span from 0% to 20% of the total number of nodes, with increments of 5%, both for the more central ("top" ablation) and the less central ("bottom" ablation) nodes. When a node is removed, all the triples containing it are removed from the KG; consequently, the number of removed triples might be higher than the number of nodes removed. Applying an ablation ratio of up to 50% can remove nearly all triples from highly connected graphs. For this reason, the maximum considered ablation is 20% to maintain a significant number of triples both in the training and in the test datasets.

Tuning. No specific hyperparameters fine-tuning was performed for the embedding models; in fact, we utilized the hyperparameters reported by [1, 3, 36] as the most effective for the LP task with the selected datasets and models.

Libraries. Pykeen [36] and iGraph [48] Python libraries have been used to store, manage, and extract metrics from the KGs (including the PageRank measure of the information loss), while the CodeCarbon [49] Python library has been used for tracking the energy consumption during the experiments.

Hardware Resources. All experiments have been performed on DGX A100 machines with Dual AMD Rome 7742 processors (128 cores), 1TB RAM, and 8x NVIDIA A100 GPUs. Each experiment has been executed allocating a single CPU core and a single GPU.

4.3 Minimal Test Set

According to the methodology described in Section 3, task performance metrics are calculated on a test set derived after a random split and controlled ablation. The collection of facts in the test set at this stage is referred to as *full test set*. However, task performance metrics calculated on different test sets – i.e., corresponding to different ablation ratios – are not necessarily comparable since they refer to different facts.

To ensure comparability across ablations, performance metrics are calculated on both the *full test set* and the *minimal test set*, the latter being fixed for each dataset split regardless of the ablation ratio. The minimal test set is defined as the intersection of all the considered test sets for a given dataset split, or equivalently, as the full test set corresponding to the maximum ablation ratio.

As ablation is performed incrementally, task performance of different models trained on differently ablated datasets remains comparable, being all evaluated against the same minimal test set.

Table 1: LP performance ratio and delta w.r.t. the original dataset performance (H@10), grouped for all experiments, after ablating the TOP (more central) and BOTTOM (less central) nodes.

DATASET	MODEL	PERFORMANCE					
		TOP			BOTTOM		
		ratio	delta	+/-	ratio	delta	+/-
WN18	TransE	0.99	-0.01		1.03	0.03	+
	ComplEx	1.01	0.01		1.04	0.03	+
	RotatE	0.97	-0.03	-	1.03	0.03	+
FB15k	TransE	1.11	0.09	+	0.99	0	
	ComplEx	1.13	0.09	+	0.99	0	
	RotatE	1	0		0.99	0	
YAGO310	TransE	0.92	-0.05	-	1.02	0.01	+
	ComplEx	1.13	0.04	+	1.02	0.01	+
	RotatE	0.77	-0.13	-	1.02	0.01	+

5 RESULTS AND DISCUSSION

In the interest of readability and conciseness, this section presents the main findings from the experiments. The full set of results is available at https://github.com/edoardoramalli/KGE_Quality.

The results highlight two main findings: *(1)* a loss in link prediction performance is never observed when ablating the less central nodes, irrespective of the test set employed, and *(2)* a significantly smaller percentage of information is lost if the ablated nodes are carefully selected, rather than chosen randomly, for model training.

These findings suggest that it is possible to reduce a KG by removing the entities, without relying on specific domain knowledge, using criteria based on node centrality, while retaining satisfactory task performance (compared to the performance observed on the original KG) and maintaining a controllable level of information in the KG. The reduced KG can lead to more effortless exploration and analysis operations, thus saving time and resources.

5.1 Impact on Task Performance

Table 1 groups, for all the dataset-model-ablation (respectively of the most central – TOP – or the least central – BOTTOM – nodes) combinations, the *ratio* and the absolute difference (*delta*) between the ablated and the original H@10 task performance, highlighting if an improvement (+) or worsening (-) of the metric is observed (column +/-). The different colors underline the cases where performance decreases with ablation (dark-red – for more than 5% and light-red – from 1% to 5%) and the cases where performance improves with the ablation (dark-green – for more than 5% and light-green – from 1% to 5%).

The results of Table 1 show that, as expected, the ablation of central nodes leads to worse or unchanged performance in most cases. The fact that some dataset-model combinations are more/less affected by the ablation may depend on various factors, such as:

(1) The density (i.e., the fraction between the number of edges and the maximal number of edges) and the transitivity (i.e., the probability that the adjacent nodes of a node are connected) of the graph: if the graph has many central nodes, many edges and loops, and fewer vertices than edges (e.g.,

FB15k), removing the most central nodes may result in unchanged or slightly improved task performance; by removing the most central nodes, the graph remains highly connected and redundant edges are removed. On the other hand, if the network is less dense, with many vertices and fewer edges (e.g., WN18), the performance deteriorates because most of the central nodes, and thus relevant connections, are removed. The YAGO310 dataset is also not particularly dense, but the number of vertices and edges is much higher than the other datasets.
(2) The tolerance of a model to the loss of information in the original KG. In fact, based on the experimental data, it seems that the ComplEx model better tolerates the ablation of central nodes.

Notably, ablating peripheral nodes (i.e., bottom ablation) often results in unchanged or even improved task performance across all observed combinations. In larger KGs (e.g., YAGO310, FB15k), the removal of peripheral nodes has little impact, while in smaller KGs (e.g., WN18), the performance improves by eliminating nodes that represent less relevant information.

As the ablation of the less central nodes does not result in a loss of task performance, we envision that ablations could be employed to save time and resources while training prediction models with KGs. For this reason, in this paper, we focus on analyzing the results related to the ablation of the least significant nodes.

5.2 Task Performance and Information Trade-off with different ablation ratios

Link prediction. Figure 2 reports the LP performance for the selected dataset-model combinations in detail: the y-axis reports the H@10 metric, averaged over three random splits, while the x-axis reports the ablation ratio. The task performance for the minimal test set is marked with triangles, while the full test set performance is marked with circles.[3] The H@10 task performance is shown in Figure 2 for each of the considered bottom ablation ratios, highlighting that the LP performance is almost unchanged with the ablation of the less central nodes, regardless of the kind of test set used. Moreover, the performance does not change remarkably when nodes are randomly removed from the KG. This suggests the ability of KGE to adapt to different subsets of the same KG and keep the same performance.

As a further motivation for using the minimal test set (see Section 4.3), Figure 3 illustrates the distribution of the differences between the task performance (H@10) on the full and the min test sets for the various ablations ratios. Most of the time, the performance does not differ significantly across the two test sets, in accordance with Figure 2. This highlights the ability of the KGE models to focus on the main content of a KG even when trained on more comprehensive data sets and tested on more focused test tests, such as the min test set. Based on this observation, we use the minimal test set to provide a more conservative and consistent performance estimate.

However, in some cases (represented by the right tails of the distribution), using the full test set may suggest a higher task performance; this happens for all the considered embedding models. Figure 2 highlights some experiments with the YAGO310 dataset as prominent examples of the difference in LP results when computed on different test sets.

Information loss. Even if a loss in prediction performance is not appreciable, applying an ablation necessarily results in a loss of information contained in the KG. For this reason, we focused on analyzing the effect of the different ablations on the amount of lost information. Section 3.5 estimates the information loss with the amount of cumulated PageRank lost w.r.t. the original dataset. As an additional measurement, we also consider the number of triples no longer present in the KG.

Figure 4 depicts the information loss extracted for the selected dataset-model combinations in more detail: The y-axis represents the percentage of information loss, while the x-axis represents the ablation ratio. The lines indicate information loss in terms of PageRank values as a proxy for information contents. Additionally, Figure 4 also shows the number of triples removed. Results are shown for the three different ablation approaches utilized (Harmonic Centrality, PageRank, and Random).

The figure highlights that using centrality for the ablation is a more performant approach, compared to random ablation, with respect to information loss. On average, across models and datasets, the random ablation results in a 5.9% and 9.3% increase in information loss compared to ablating with Harmonic Centrality and PageRank, respectively (the maximum performance difference is 10.1% and 15.6%, respectively). This comparison highlights that the amount of retained information is influenced by the metric utilized to perform the ablation. The amount of information lost for different ablation ratios depends on the KG characteristics. Smaller KGs (e.g., WN18) have a higher probability of losing information.

Given the above considerations, we argue that properly crafted ablations could be used as a means of saving computational resources, and thus energy consumption, while balancing the loss of information that occurs with removing nodes in a KG. The next section proposes a set of ML models to *(i)* estimate how much energy can be saved by employing ablations before training, *(ii)* recommending the number of nodes triples and which centrality measure to use for ablating a given KG to *(iii)* minimize the information loss.

5.3 Impact of Ablation on Sustainability

Training a KGE model requires a significant amount of computational resources. The experiments described in Section 4 require the usage of a high-end GPU for an extended amount of time, which substantially affects both cost and energy consumption. For instance, running a single experiment on the cheapest AWS GPU (using *g4dn.xlarge* instance type)[4] for 5 hours (longest experiment we executed) costs ~2.63 USD and consumes ~0.52 kWh.[5]

For all the experiments described in Section 4, we measured the total training duration (in seconds) and estimated the energy consumption (in kWh).

[3]Given the high computational cost and limited availability of resources, the random ablation results were computed only for one model.

[4]https://aws.amazon.com/it/ec2/instance-types/g4/

[5]https://engineering.teads.com/sustainability/carbon-footprint-estimator-for-aws-instances/

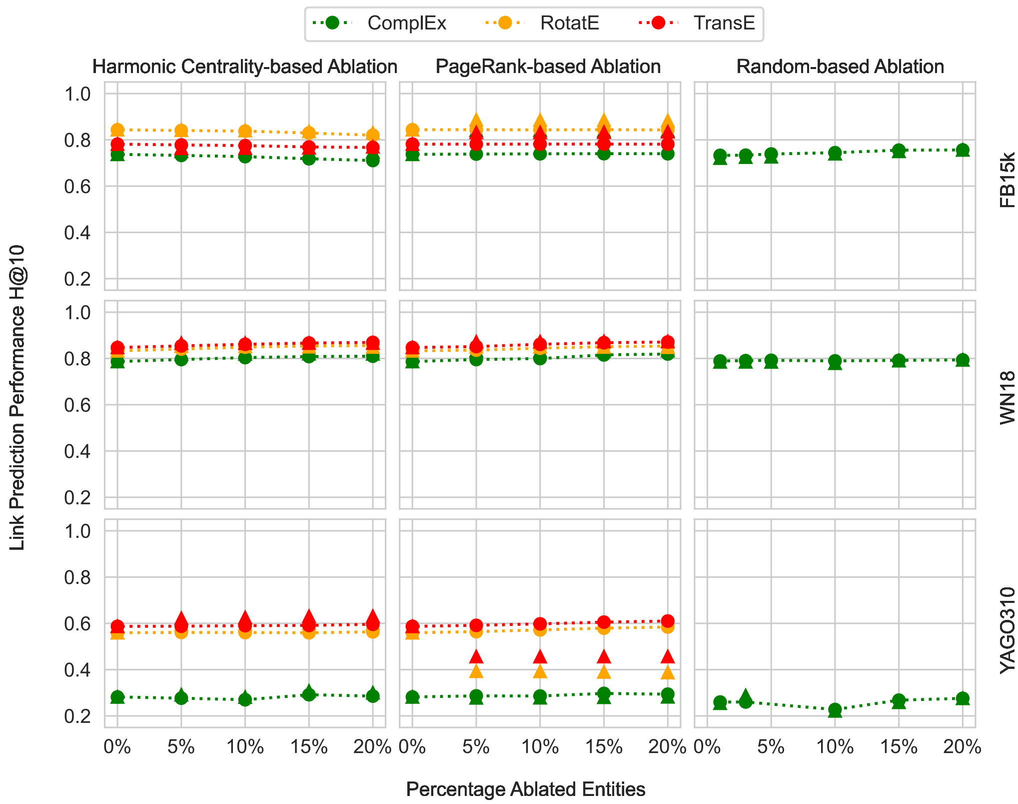

Figure 2: Link prediction performance for FB15k, WN18, and YAGO310 datasets, average over three independent splits. Circles indicate results for the full test set and triangles for the minimal test set.

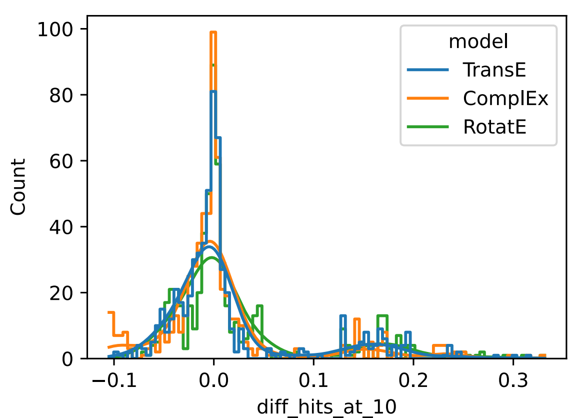

Figure 3: Distribution of the LP performance difference (H@10) between full and minimal test set.

We consider only the energy consumption of the GPU. The energy consumed by the CPU is negligible because we verified that *(i)* the CodeCarbon library measures the maximum instantaneous CPU energy consumption instead of the actual one, and *(ii)* during the model training, the CPU is almost inactive.

Analyzing the results, the training time significantly changes according to the size of the dataset and the model considered. Figure 5 shows the dependency between the number of entities in the graph (x-axis), the training time (y-axis), the model, and the dataset employed, with the time related to the original graphs indicated with a cross. It can be observed that model training can require several hours for KGs with a large number of entities (almost 5 hours for the largest graph, YAGO310). Moreover, the training time is significantly affected by the number of entities: it is evident that as the KG's entity size increases, the training time increases; the extent of this relation depends on the dataset and link prediction model considered.

6 IMPROVING SUSTAINABILITY WITH NODE ABLATION

In this section, we propose possible ways to exploit the results and insights obtained from our experiments. In particular, we analyze how we can use ablation to reduce the cost and/or the energy consumption of the KG training phase while keeping the estimated quality level within a required threshold.

Considering that: *(i)* training time can be considerably reduced ablating the graph, and *(ii)* it has been shown in Section 5 that LP performance is not considerably affected by ablating the least

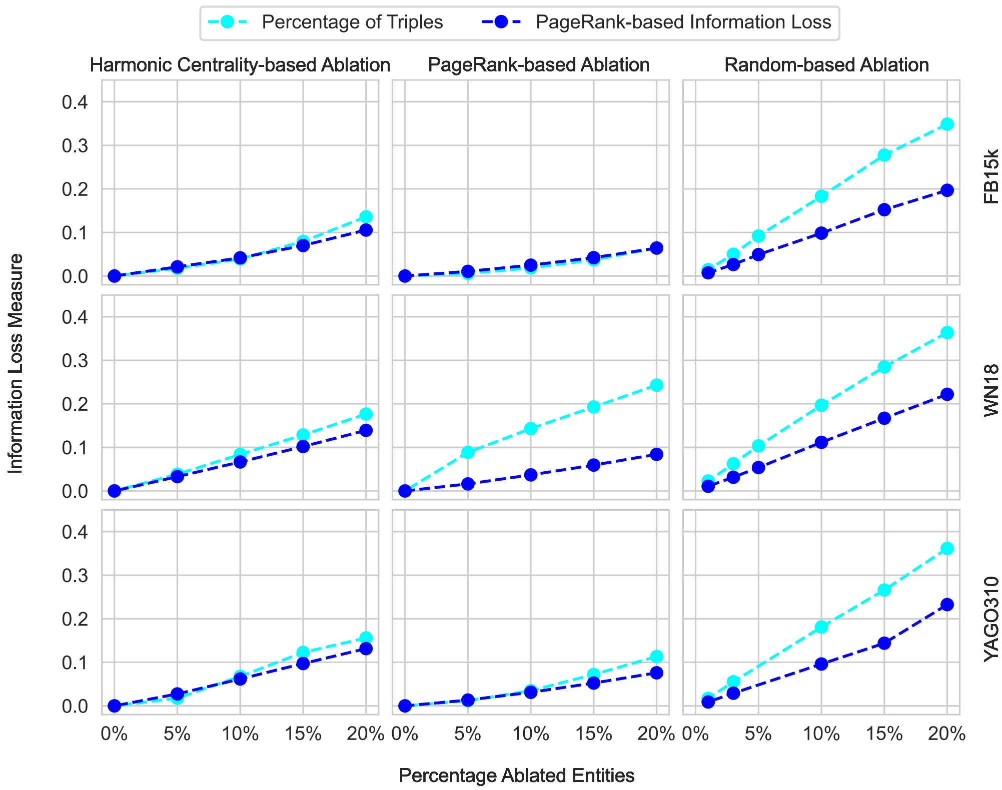

Figure 4: Information loss in percentage, measured in terms of PageRank, and number of triples removed

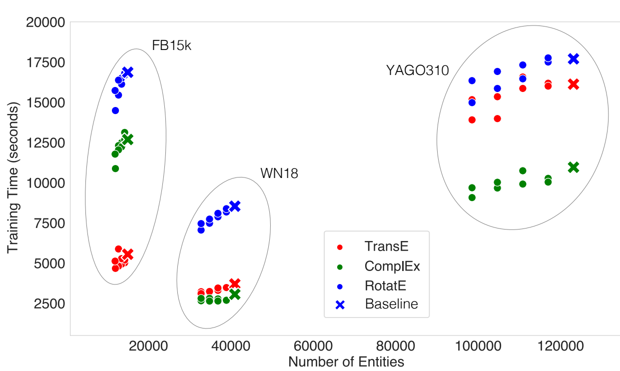

Figure 5: Model training time w.r.t. the number of entities used for training

central nodes, KG ablation can be used for reducing costs and/or energy, while still achieving good performance.

To this end, in this section, we propose a predictive model to estimate the number of entities that can be removed from the graph while maintaining a satisfactory quality level, taking into consideration the maximum information loss as a specified requirement.

6.1 Prediction Models for Ablation Ratio and Training Time

As discussed in Section 5.3, ablation can save costs and energy by significantly reducing the Training Time (TT) where the same amount of computational resources is available. However, we have also seen how the number of ablated nodes affects the loss of information, i.e., removing nodes results in a loss of relevant information. Thus, the preferred trade-off between information and TT needs to be determined.

In this section, we aim to balance this trade-off by employing machine-learning models to suggest (1) a suitable number of nodes that can be ablated while maintaining a certain information level, (2) which centrality measure is the most convenient to use for entity ablation, and (3) an estimate of how much the training time can be reduced and, consequently, what is the amount of cost and energy that can be saved.

We start by assuming that a user specifies a threshold for the information that is acceptable to be lost (the $InfLoss$, as stated in Section 3.5.2). Subsequently, the Percentage of Information (PoI), i.e., the sum of the original PageRank values of the remaining nodes after the ablation, that must be preserved is:

$$PoI = 1 - InfLoss \quad (3)$$

We train two different models for each centrality measure, used as a basis for ablation (i.e., Harmonic Centrality and PageRank):

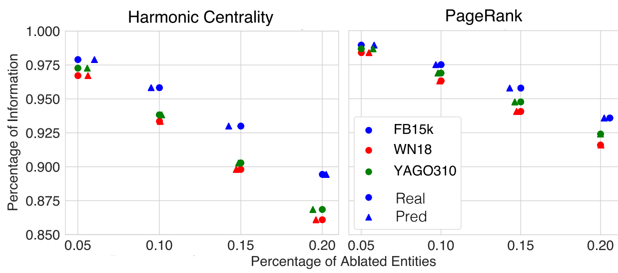

Figure 6: Real and predicted values for *Model 1*. Relation between *Percentage of Information* and *Percentage of Ablated Entities* with the two ablation methods

Model 1. Given the required PoI, the goal of this first model is to estimate the *maximum number of entities (nodes of the KG) that can be removed* while ensuring the PoI set by the user.

Model 2. Given a number of entities in an ablated KG, the second model estimates the *time reduction* for training a specific embedding model with the ablated KG compared with the full set of entities.

Using the prediction results of *Model 1* and *Model 2* (i.e., the estimated number of entities and saved training time) for each centrality measure for ablation, we can suggest the best centrality measure to adopt to perform the most time-saving ablation. Finally, we aim to provide the user with an estimation of costs and energy savings given the new training time.

To build the models, we transformed the number of entities and the training time into percentage values compared to their baseline values (i.e., the entities and TT of the original KG). As depicted in Section 3.5.3, we calculate the Percentage of Ablated Entities and the Percentage of Time Saved (PTS) measures.

Training & Testing. To train and test the models, we perform the *leave-one-dataset-out* process: for each analyzed dataset d, we train the regression models with all the data collected from the experiments, excluding the results associated with that dataset (*training set*: $[X_{d-}, y_{d-}]$). Then, we use the results associated with the dataset d to test the models (*testing set*: $[X_{d+}, y_{d+}]$).

Model 1. A linear relation between the PoI loss and the number of entities can be appreciated in Figure 4. For this reason, we employ a linear regression model that takes as input the PoI (X_{PoI}) and predicts an estimated value for the entities (y_e). Figure 6 represents the actual values and those predicted by the regressor (using the *leave-one-dataset-out* process), respectively, with circles and triangles, for each of the three datasets. Predictions are very close to the actual values for all three analyzed datasets; in most cases, the number of entities to ablate is slightly underestimated by the regressor, which is beneficial for not exceeding the threshold of PoI after ablation. Moreover, it can be appreciated that, for the same amount of PoI, using PageRank for selecting nodes allows for the ablation of a higher percentage of entities than with Harmonic Centrality, compatibly with the observations proposed in Section 5.2.

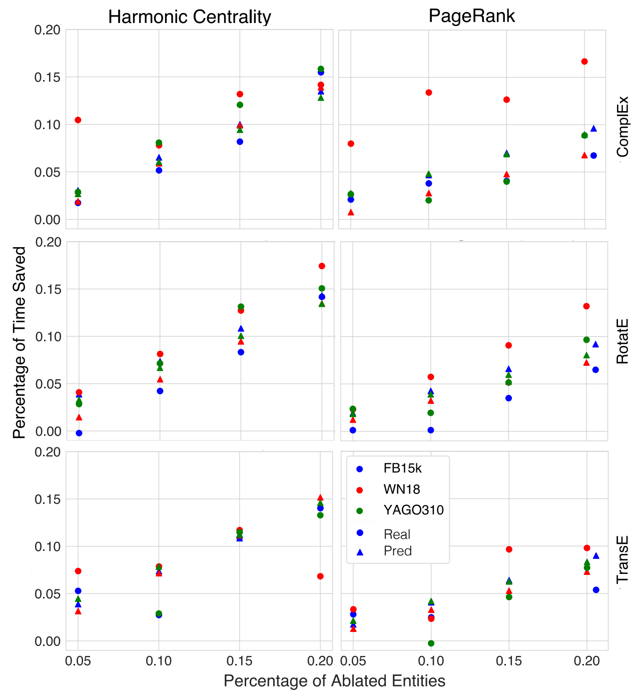

Figure 7: Real and predicted values for *Model 2*. Relation between *Percentage of Training Time Saved* and *Percentage of Ablated Entities* for the three embedding models

Model 2. Figure 5 shows that, having fixed the dataset and the KGE model, there is a roughly linear relation between the number of entities and the TT. Similarly to *Model 1*, we train a linear regression model that takes as input the number of entities (X_e) and predicts an estimated value for the Percentage of Saved TT (y_{PTS}) with each given embedding technique. Figure 7 also reports the actual and the predicted (using the *leave-one-dataset-out* process) values with circles and triangles. Predicting the saved training time is more complex; estimated values for the WN18 dataset are not very accurate for all three embedding models, whereas predictions for YAGO are not accurate for TransE. In terms of saved time, Figure 7 shows that Harmonic Centrality-based ablation generally saves more time than PageRank-based ablation.

This draws attention to a trade-off: using PageRank-based ablation favors PoI while using Harmonic Centrality-based ablation favors Training Time-saving.

Once trained and validated, the models can be used to estimate the TT required when performing the ablation with the two different centrality measures. The outcome of the prediction can be used to suggest how many entities to ablate and which centrality measure to use for ablation to get the lower value of estimated TT ensuring a required PoI.

6.2 Validation

Performance results of the *leave-one-dataset-out* procedure are displayed in Figure 8. The tables show the Mean Absolute Error (MAE)

		MAE		RMSE	
	Test set	Harmonic Centrality	PageRank	Harmonic Centrality	PageRank
Model 1 Utility Loss → Entities	FB15k	0.006250	0.005461	0.000047	0.000034
	WN18	0.003332	0.002136	0.000015	0.000008
	YAGO310	0.003602	0.003297	0.000018	0.000018
Model 2 Entities → Saved Time	FB15k	0.019024	0.027550	0.000562	0.001145
	WN18	0.033581	0.048999	0.001801	0.003437
	YAGO310	0.021580	0.018131	0.000781	0.000537

(a)

Percentage of:	*Ablated Entities*	*Training Time Saved*
max	0.205873	0.213705
average	0.111282	0.060525
stddev	0.066149	0.051899

(b)

Figure 8: (a) Root Mean Squared Error obtained by testing *Model 1*, and *Model 2* using the *leave-one-dataset-out* procedure; (b) ground truth distribution for *Percentage of Ablated Entities* and *Percentage of Training Time Saved*

and the Root Mean Squared Error (RMSE) obtained by each regression model, excluding that dataset from the training set and using it for testing. Moreover, the distribution of the real values of the target variables is reported below. Given the linear relation between all the features involved, we obtained good performance for both the developed models.

The predicted computational time can be used to estimate cost, energy consumption, and environmental impact associated with the training task.

The cost C can be computed as:

$$C = y_{TT} * cost_h \tag{4}$$

where y_{TT} is the estimated training time and $cost_h$ is the hourly cost of the computational instance used for the training.

The energy E can be computed as:

$$E = y_{TT} * P \tag{5}$$

where P is the average power consumption of the computational instance used for the training.

The carbon emissions CE generated for the training can be computed as:

$$CE = E * GHG \tag{6}$$

where GHG is the Greenhouse Gas emission intensity[6] of the computational instance used for the training.

Running example. We can imagine starting from a dataset such as YAGO310, which has 123,143 entities and 17,698 seconds of training time for RotatE, ∼5 hours, and that the user set the PoI at 90%.

[6]https://www.eea.europa.eu/en/analysis/indicators/greenhouse-gas-emission-intensity-of-1

According to Model 1, the percentage of entities that can be ablated maintaining the required constraint using Harmonic Centrality is approximately 15%, with a new predicted training time ∼90% of the original one with Model 2 (considering our experiments, the real new training time is 88.52% of the original one). On the other hand, for PageRank, the number of entities that can be ablated is 20%, but the training time will be no less than ∼92% w.r.t. the original.

For this reason, the Harmonic Centrality measure will be suggested to the user to ablate the 15% of the original entities, with a new estimated TT of 15,928 seconds, ∼4.424 hours, and a total cost of ∼2.33 USD using a *g4dn.xlarge* instance. The cost reduction can be estimated as 0.30 USD and the energy reduction is 60 Wh (from 0.52 kWh to 0.46 kWh) for a single training. In relative numbers, the total improvement in terms of costs, time, and energy is 11%.

7 CONCLUDING REMARKS

In this paper, we have conducted an experimental evaluation of various KG ablation strategies, observing and measuring their effect on the quality of Link Prediction (LP). We used a domain-independent approach, testing different embedding algorithms on diverse datasets. The results of our study led us to the conclusion that ablating the less central nodes allow us to achieve good performance while maintaining the KG semantic integrity (i.e., information loss). The proposed approach can be exploited to facilitate LP sustainability, positively affecting both the cost and the energy consumption of the model training. Finally, we used our results to develop ML models to recommend the best sustainability-aware ablation strategy, based on the required threshold on information loss set by the user.

ACKNOWLEDGMENTS

This work has been supported by the PRIN 2022 Project “Discount quality for responsible data science: Human-in-the-Loop for quality data”, by the PNRR-PE-AI “FAIR” project funded by the NextGenerationEU program, and by the NRF Korea, Grant Number 2022R1F1A1072843. We acknowledge the CINECA award under the ISCRA initiative for the availability of high-performance computing resources and support.

REFERENCES

[1] Ilaria Ferrari, Giacomo Frisoni, Paolo Italiani, Gianluca Moro, and Claudio Sartori. Comprehensive analysis of knowledge graph embedding techniques benchmarked on link prediction. *Electronics*, 11(23):3866, 2022.
[2] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in Neural Information Processing Systems*, 33:22118–22133, 2020.
[3] Andrea Rossi, Denilson Barbosa, Donatella Firmani, Antonio Matinata, and Paolo Merialdo. Knowledge graph embedding for link prediction: A comparative analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(2):1–49, 2021.
[4] Yike Liu, Tara Safavi, Abhilash Dighe, and Danai Koutra. Graph summarization methods and applications: A survey. *ACM Computing Surveys*, 51(3):1–34, 2018.
[5] Qi Song, Yinghui Wu, Peng Lin, Luna Xin Dong, and Hui Sun. Mining summaries for knowledge graph search. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1887–1900, 2018.
[6] Yi Wang, Ying Wang, and Qia Wang. Scalable summarization for knowledge graphs with controlled utility loss. In *International Conference on Database and Expert Systems Applications*, pages 21–36. Springer, 2023.

[7] Matteo Lissandrini, Torben Bach Pedersen, Katja Hose, and Davide Mottin. Knowledge graph exploration: where are we and where are we going? *ACM SIGWEB Newsletter*, 2020(Summer):1–8, 2020.

[8] Ali Yassin, Abbas Haidar, Hocine Cherifi, Hamida Seba, and Olivier Togni. An evaluation tool for backbone extraction techniques in weighted complex networks. *Scientific Reports*, 13(1), 2023.

[9] Qingxia Liu, Gong Cheng, Kalpa Gunaratna, and Yuzhong Qu. Entity summarization: State of the art and future challenges. *Journal of Web Semantics*, 69:100647, 2021.

[10] Yike Liu, Tara Safavi, Abhilash Dighe, and Danai Koutra. Graph summarization methods and applications: A survey. *ACM Computing Surveys*, 51(3), 2018.

[11] Šejla Čebirić, François Goasdoué, Haridimos Kondylakis, Dimitris Kotzinos, Ioana Manolescu, Georgia Troullinou, and Mussab Zneika. Summarizing semantic graphs: a survey. *The VLDB Journal*, 28:295–327, 2019.

[12] Matteo Riondato, David García-Soriano, and Francesco Bonchi. Graph summarization with quality guarantees. *Data Mining and Knowledge Discovery*, 31:314–349, 2017.

[13] Caleb Belth, Xinyi Zheng, Jilles Vreeken, and Danai Koutra. What is normal, what is strange, and what is missing in a knowledge graph: Unified characterization via inductive summarization. In *Proceedings of The Web Conference 2020*, pages 1115–1126, New York, NY, 2020. ACM.

[14] Zeqian Shen, Kwan-Liu Ma, and T. Eliassi-Rad. Visual analysis of large heterogeneous social networks by semantic and structural abstraction. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1427–1439, 2006.

[15] Daniel A. Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.

[16] Joshua Batson, Daniel A. Spielman, Nikhil Srivastava, and Shang-Hua Teng. Spectral sparsification of graphs: theory and algorithms. *Commun. ACM*, 56(8):87–94, 2013.

[17] Paolo Boldi and Sebastiano Vigna. Axioms for centrality. *Internet Mathematics*, 10(3-4):222–262, 2014.

[18] K. Ashwin Kumar and Petros Efstathopoulos. Utility-driven graph summarization. *Proc. VLDB Endow.*, 12(4):335–347, 2018.

[19] Mahdi Hajiabadi, Jasbir Singh, Venkatesh Srinivasan, and Alex Thomo. Graph summarization with controlled utility loss. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 536–546, 2021.

[20] Tara Safavi, Caleb Belth, Lukas Faber, Davide Mottin, Emmanuel Müller, and Danai Koutra. Personalized knowledge graph summarization: From the cloud to your pocket. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 528–537. IEEE, 2019.

[21] Chong Cher Chia, Maksim Tkachenko, and Hady W. Lauw. Semantically constitutive entities in knowledge graphs. In *Database and Expert Systems Applications: 34th International Conference, DEXA 2023*, pages 445–461, Berlin, Heidelberg, 2023. Springer-Verlag.

[22] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality assessment for linked data: A survey. *Semantic Web*, 7(1):63–93, 2016.

[23] Bingcong Xue and Lei Zou. Knowledge graph quality management: a comprehensive survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4969–4988, 2022.

[24] Cinzia Cappiello, Tommaso Di Noia, Bogdan Alexandru Marcu, and Maristella Matera. A quality model for linked data exploration. In *Web Engineering: 16th International Conference, ICWE 2016*, pages 397–404. Springer, 2016.

[25] Ruiqi Zhu, Alan Bundy, Jeff Pan, Kwabena Nuamah, Fangrong Wang, Xue Li, Lei Xu, and Stefano Mauceri. Assessing the quality of a knowledge graph via link prediction tasks. In *Proceedings of the 2023 7th International Conference on Natural Language Processing and Information Retrieval*, pages 124–129, 2023.

[26] Budhitama Subagdja, D Shanthoshigaa, Zhaoxia Wang, and Ah-Hwee Tan. Machine learning for refining knowledge graphs: A survey. *ACM Computing Surveys*, 56(6):1–38, 2024.

[27] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017.

[28] Jiahang Cao, Jinyuan Fang, Zaiqiao Meng, and Shangsong Liang. Knowledge graph embedding: A survey from the perspective of representation spaces. *ACM Computing Surveys*, 56(6):1–42, 2024.

[29] Edoardo Ramalli, Alberto Parravicini, Guido W Di Donato, Mirko Salaris, Céline Hudelot, and Marco D Santambrogio. Demystifying drug repurposing domain comprehension with knowledge graph embedding. In *2021 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 1–5. IEEE, 2021.

[30] Yu Li, Zhu-Hong You, Shu-Min Wang, Cheng-Gang Mi, Mei-Neng Wang, Yu-An Huang, and Hai-Cheng Yi. Attention-based learning for predicting drug-drug interactions in knowledge graph embedding based on multisource fusion information. *International Journal of Intelligent Systems*, 2024(1):5155997, 2024.

[31] Meihong Wang, Linling Qiu, and Xiaoli Wang. A survey on knowledge graph embeddings for link prediction. *Symmetry*, 13(3):485, 2021.

[32] Mohamad Zamini, Hassan Reza, and Minou Rabiei. A review of knowledge graph completion. *Information*, 13(8):396, 2022.

[33] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.

[34] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Mikhail Galkin, Sahand Sharifzadeh, Asja Fischer, Volker Tresp, and Jens Lehmann. Bringing light into the dark: A large-scale evaluation of knowledge graph embedding models under a unified framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):8825–8845, 2021.

[35] Sameh K. Mohamed, Emir Muñoz, and Vit Novacek. On training knowledge graph embedding models. *Information*, 12(4):147, 2021.

[36] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Sahand Sharifzadeh, Volker Tresp, and Jens Lehmann. Pykeen 1.0: a python library for training and evaluating knowledge graph embeddings. *Journal of Machine Learning Research*, 22(82):1–6, 2021.

[37] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[38] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*, 2019.

[39] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pages 2071–2080. PMLR, 2016.

[40] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2):494–514, 2022.

[41] Massimo Marchiori and Vito Latora. Harmony in the small-world. *Physica A: Statistical Mechanics and its Applications*, 285(3):539–546, 2000.

[42] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford infolab, 1999.

[43] Norbert Fuhr. Some common mistakes in IR evaluation, and how they can be avoided. *SIGIR Forum*, 51(3):32–41, 2018.

[44] Tetsuya Sakai. On Fuhr's guideline for IR evaluation. *SIGIR Forum*, 54(1), Feb. 2021.

[45] Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD Conference*, pages 1247–1250. ACM, 2008.

[46] Christiane Fellbaum. *WordNet: An electronic lexical database.* MIT press, 1998.

[47] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: a core of semantic knowledge. In *WWW*, pages 697–706. ACM, 2007.

[48] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006.

[49] Benoit Courty, Victor Schmidt, Sasha Luccioni, Goyal-Kamal, MarionCoutarel, Boris Feld, Jérémy Lecourt, LiamConnell, Amine Saboni, Inimaz, supatomic, Mathilde Léval, Luis Blanche, Alexis Cruveiller, ouminasara, Franklin Zhao, Aditya Joshi, Alexis Bogroff, Hugues de Lavoreille, Niko Laskaris, Edoardo Abati, Douglas Blank, Ziyao Wang, Armin Catovic, Marc Alencon, Michał Stęchły, Christian Bauer, Lucas-Otavio, JPW, and MinervaBooks. mlco2/codecarbon: v2.4.1, 2024.