ToT4ES: Tree of Thought for Entity Summarization

Asep Fajar Firmansyah[0000−0001−5775−6966], Hamada M. Zahera[0000−0003−0215−1278], and Axel-Cyrille Ngonga Ngomo[0000−0001−7112−3516]

Data Science Group (DICE), Heinz Nixdorf Institute, Department of Computer Science, Paderborn University
{asep.fajar.firmansyah, hamada.zahera, axel.ngonga}@upb.de

Abstract. Entity summarization aims to generate concise summaries of entities by selecting a size-constrained subset of RDF triples from an entity's description. This process reduces cognitive overload and enables users to comprehend the essential information more efficiently. Existing approaches mainly rely on supervised learning, which requires large volumes of labeled data. To overcome these challenges, this study introduces an extractive method using large language models (LLMs), termed ToT4ES (**T**ree **o**f **T**hought for **E**ntity **S**ummarization). ToT4ES is a training-free framework leveraging the tree of thought strategy to produce high-quality entity summaries without any task-specific fine-tuning. The summarization task is decomposed into three complementary objectives: relatedness, informativeness, and coverage/diversity. We employ a tree search strategy to explore potential summaries. At each depth level, candidate partial summaries (states) are evaluated using an LLM-guided heuristic score. This score is a weighted combination of our three objectives. The search then prunes the tree to retain only the most promising candidate summaries. The proposed approach is evaluated on the ESBM benchmark (DBpedia, LinkedMDB) and the FACES datasets, considering two summary size constraints ($k = 5$ and $k = 10$). Experimental results indicate that ToT4ES outperforms existing unsupervised methods based on F-measure. Notably, ToT4ES achieves a 20.3% relative improvement on LinkedMDB. Our implementation, including source code and datasets, is available at https://github.com/dice-group/ToT4ES.

Keywords: Entity Summarization · Large Language Models · Tree of Thought

1 Introduction

The rapid expansion of knowledge graphs (KGs) across diverse domains, from general ones like DBpedia to specific domains such as healthcare [1], education [12], and finance [13], has led to a substantial increase in the volume of structured data [11,21]. Consequently, the sheer amount of information linked to a single entity can easily overwhelm users and hinder their ability to identify salient

facts [23]. This challenge motivates the task of entity summarization, which aims to select a compact and relevant subset of RDF triples that preserves essential information while reducing cognitive load [16]. For instance, entity summarization enables the RDF browser [5] to present a concise RDF representation of entities on the web, facilitating user comprehension. Existing entity summarization methods generally fall into two categories [18]: Unsupervised methods that rely on rigid, hand-crafted heuristics [3,9,23,27], or supervised methods that require large volumes of labeled training data, inherently limiting their transferability across domains [6,7,19,31]. Furthermore, most traditional methods frame this task as a simple "learning-to-rank" problem [10], evaluating and scoring triples in isolation rather than considering the set-level coherence of the final summary. To overcome these limitations, we formulate extractive entity summarization as a Tree-of-Thought-guided, multi-trajectory subset search with iterative state expansion and beam pruning under weighted semantic constraints. We introduce ToT4ES (**T**ree **o**f **T**hought for **E**ntity **S**ummarization), a training-free framework that uses the reasoning abilities of Large Language Models (LLMs) to dynamically navigate the summary search space. Specifically, ToT4ES formulates selection as a multi-step Think-Branch-Summarize planning process. In contrast to standard beam search for token-level sequence decoding, it maintains a beam of state trajectories over candidate sub-summaries. The LLM acts both as generator (Think) and critic (Summarize): Think/Branch propose and expand partial states, while Summarize evaluates them along relatedness, informativeness, and coverage/diversity for beam pruning. This tree-search framework prunes suboptimal paths and mitigates greedy local traps, overcoming single-pass LLM rankings that score triples in isolation and ignore cross-triple redundancy. This approach enforces strict size bounds and aims to optimize the final triple subset jointly across relatedness, informativeness, and coverage/diversity. Our results demonstrate that ToT4ES consistently outperforms state-of-the-art unsupervised methods, achieving gains of up to 20.3% in F-measure on the LinkedMDB dataset, while also surpassing LLM baselines across five out of six benchmark settings. The main contributions are summarized as follows:

- We introduce ToT4ES, a training-free framework that adapts LLMs through a Tree of Thought search strategy for extractive entity summarization.
- We introduce a deliberative search process in ToT4ES that comprises three functional components: Think (which encompasses three specific subtasks), Branch, and Summarize. This process explicitly optimizes for key semantic criteria, including relatedness, informativeness, and coverage/diversity.
- We provide an extensive empirical evaluation across three standard benchmarks (DBpedia, LinkedMDB, and FACES), including an ablation study and hyperparameter sensitivity analysis.
- The implementation is released as open-source code, including prompt factories and the modular search engine, to facilitate further research in LLM-driven knowledge graph summarization.

2 Related Work

Unsupervised Learning Entity Summarization. In unsupervised entity summarization, methods typically rely on hand-crafted features—used individually or in combination—such as relatedness (frequency and centrality), informativeness, and diversity/coverage to select the RDF triples most relevant to an entity [18]. For instance, RELIN [3] ranks an entity's triples by combining a random-surfer relatedness centrality (weighted PageRank) with a corpus-based informativeness measure, then selects the top k (e.g., $k = [5, 10]$) triples. Unlike RELIN, which combines relatedness and informativeness features, LinkSUM [26] is relatedness-oriented: it combines centrality features—such as PageRank—with backlink signals for resource selection and uses property-frequency statistics for relation selection. By contrast, FACES [9] explicitly incorporates diversity when selecting facts for the entity summary. Furthermore, BAFREC [16] balances predicate frequency and rarity by utilizing ontology-derived statistics to optimize relatedness and informativeness. MPSUM [30] applies predicate-based matching and Latent Dirichlet Allocation (LDA) topic modeling to identify thematically coherent triples. KAFCA [33] employs Formal Concept Analysis to group concepts and select representative facts. More recently, IRES [20] adopts graph neural methods to model indirect relationships, leveraging multi-hop context to enhance summarization. Collectively, these baselines illustrate complementary strategies in frequency and rarity, topic structure, concept lattices, and relational reasoning, providing a comparative framework for evaluating ToT4ES's multi-dimensional, search-based approach.

Supervised Entity Summarization. In contrast, recent supervised approaches formulate entity summarization as a learning-to-rank problem and employ neural models with learned embeddings to score triples. These models are trained on labeled data (e.g., the ESBM benchmark [17] and FACES [9]) and then used to select the highest-scoring subset as the entity summary. ESA [31] is the first deep-learning approach to entity summarization, formulating the task as a learning-to-rank problem with a BiLSTM and supervised attention. It encodes predicates with learned embeddings, such as word2vec [2], and objects with TransE [29] embeddings, consuming predicate–object pairs to score and rank triples. Unlike ESA, DeepLENS [19] leverages a BiLSTM architecture with a permutation-invariant, text-semantic model. It represents predicates and values using pre-trained text embeddings (e.g., fastText [14]), encodes triples with a multilayer perceptron (MLP), aggregates all triples in the entity description via attention to obtain a context vector, and finally scores each candidate triple with an MLP conditioned on that context. Furthermore, GATES [6] introduces graph neural networks—specifically graph attention networks (GATs) [28]—to the entity summarization task. By contrast, ESLM [7] leverages pretrained contextual language models (e.g., BERT [4], ERNIE [25], T5 [24]) to encode triple text and combines these representations with an attention mechanism and an MLP to rank triples. While these supervised approaches advance the state of the art in the entity summarization task, they rely on substantial amounts of labeled data.

This challenge becomes particularly pronounced when implementing such models in new domains, leading to poor out-of-domain transferability and necessitating costly relabeling efforts to adapt [20].

Language Model-Driven Entity Summarization The integration of language models into entity summarization has substantially advanced the field by enabling the capture of complex semantic patterns beyond those present in raw, structured data. Early supervised extractive methods, such as ESA [31], DeepLENS [19], and GATES [6], employed static word representations (e.g., word2vec [2], FastText [14], and GloVe [22]) to enhanced triple embeddings. To address the limitations of static embeddings, ESLM [7] introduced pre-trained encoder-decoder language models that leverage T5 to capture deeper contextual dependencies. More recently, in the abstractive domain, ANTS [8] utilized large language models (such as GPT-4) to generate implicit, absent triples, thereby addressing informational gaps and enriching the resulting summaries. While language models have mainly dominated supervised and generative-abstractive pipelines, the present work diverges from these approaches. Instead of relying on data-intensive supervision or unconstrained text generation, ToT4ES introduces an unsupervised, training-free framework that leverages LLM reasoning to dynamically select a semantically coherent subset of factual triples. By shifting the task from data-driven pattern matching to generalized semantic reasoning, this framework overcomes the data-dependency bottleneck, thereby improving its transferability across different domains without requiring retraining data labels.

3 Our Approach

3.1 Preliminaries

Let $\mathcal{E}$, $\mathcal{R}$, $\mathcal{C}$, and $\mathcal{L}$ denote the sets of entities, relations, classes, and literals, respectively. A KG is defined as a set of triples:

$$\mathcal{T} \subseteq \mathcal{E} \times \mathcal{R} \times (\mathcal{E} \cup \mathcal{C} \cup \mathcal{L}), \tag{1}$$

where each $(s, r, o) \in \mathcal{T}$ asserts that the relation r holds between the subject s and the object o. Graph-theoretically, a KG is modeled as a directed, labeled multi-graph [11]. In this graph, entities ($\mathcal{E}$) and classes ($\mathcal{C}$) serve as resource nodes, relations represent labeled directed edges, and literals ($\mathcal{L}$) function strictly as terminal nodes with no outgoing edges.

Entity Description Each entity $e \in \mathcal{E}$ is associated with an entity description, denoted by $\text{Desc}(e, \mathcal{T})$, which comprises the set of triples in $\mathcal{T}$ where e appears as either the subject or the object. Formally, this set is defined as:

$$\text{Desc}(e, \mathcal{T}) = \{(s, r, o) \in \mathcal{T} \mid s = e \vee o = e\}. \tag{2}$$

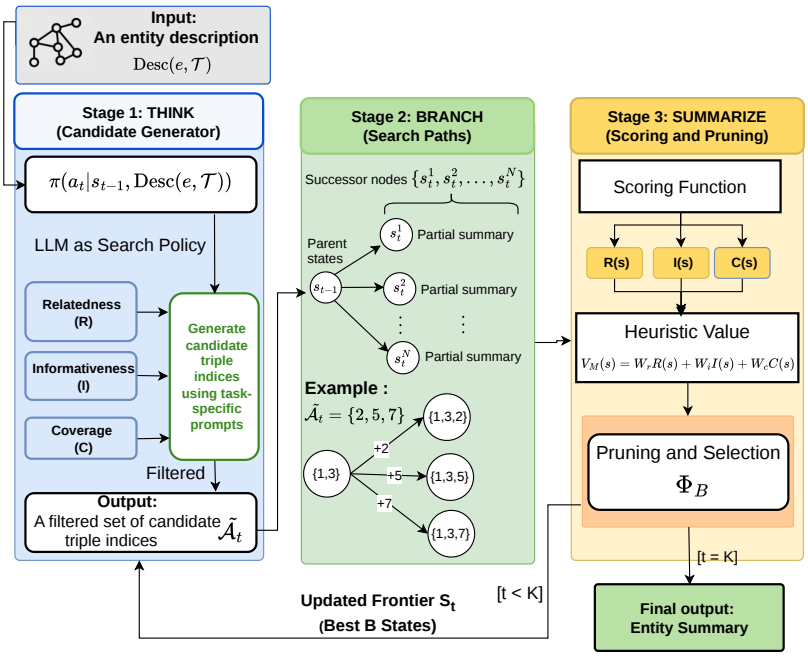

Fig. 1: The architecture Tree of Thought for entity summarization (ToT4ES)

Entity Summarization Given an entity description $\mathrm{Desc}(e, \mathcal{T})$, the task of entity summarization is to generate an optimal summary [18], denoted by S_e^*. Formally, we seek a candidate subset $S \subseteq \mathrm{Desc}(e, \mathcal{T})$ that maximizes a quality objective under a size constraint k:

$$S_e^* = \text{find} \underset{S \subseteq \mathrm{Desc}(e,\mathcal{T})}{\arg\max} \ \mathrm{score}(S \mid \mathcal{T}), \quad \text{subject to } |S| \leq k. \tag{3}$$

Here, $\mathrm{score}(S \mid \mathcal{T})$ denotes the quality score of a candidate triple subset. In our formulation, these scores are defined through a ranking objective that balances factual entity-relatedness, factual informativeness, and structural coverage/diversity.

3.2 The Architecture

Figure 1 illustrates the architecture of ToT4ES, which consists of three distinct stages—*Think, Branch, and Summarize*. The model incrementally constructs the final entity summary by adding one triple at each step. This process runs for at most K steps, where the maximum summary length is $L_{max} = K$. At each step $k \in \{1, 2, \ldots, L_{max}\}$, one triple is selected from the remaining candidates.

Stage 1: Think (Candidate Generator) In the *Think* stage, the framework generates candidate next-step expansions by querying *three task-specific* LLM policies (Relatedness, Informativeness, and Coverage/Diversity), rather than a single monolithic selector. Formally, we use a unified policy notation:

$$\pi(a_t \mid s_{t-1}, \mathrm{Desc}(e, \mathcal{T})), \tag{4}$$

where π is shorthand for the task-decomposed policies. At depth t, each policy proposes an index from the feasible action set:

$$\mathcal{A}_t = \mathrm{Idx}(\mathrm{Desc}(e, \mathcal{T}) \setminus s_{t-1})\,. \tag{5}$$

The three task-specific policies are:

$$\pi_R(a_t \mid s_{t-1}, \mathrm{Desc}(e, \mathcal{T})), \quad \pi_I(a_t \mid s_{t-1}, \mathrm{Desc}(e, \mathcal{T})), \quad \pi_C(a_t \mid s_{t-1}, \mathrm{Desc}(e, \mathcal{T})),$$

with $a_t \in \mathcal{A}_t$.

Let $\mathcal{S}_R^t, \mathcal{S}_I^t, \mathcal{S}_C^t \subseteq \mathcal{A}_t$ denote the candidate sets sampled from (π_R, π_I, π_C) at depth t. Here, a_t is the discrete action of selecting one triple index at depth t. Each policy conditions on: (i) the full description $\mathrm{Desc}(e, \mathcal{T})$ (global context), (ii) the current partial summary state (s_{t-1}), and (iii) a task-specific instructional prompt.

The three semantic criteria are handled independently during candidate generation:

- *Relatedness* (R)*:* prioritizes triples that are central and semantically defining for the entity.
- *Informativeness* (I)*:* prioritizes specific, non-trivial facts while down-weighting generic or high-frequency predicates.
- *Coverage/Diversity* (C)*:* prioritizes triples that add novel facets and reduce redundancy with (s_{t-1}).

Box 3.2 provides a streamlined excerpt of the evaluation prompt for the entity *Adele*. This structure forces the LLM to ground its candidate selection in predicate frequency and value specificity rather than generic text generation.

Example: Instructional Prompt Structure for Candidate Generation (Relatedness Domain)

Role: You are evaluating triples for RELATEDNESS to the entity.
Target Entity: Adele
Relatedness Definition:

1. *Centrality:* Uses core/frequent predicates that define entity types.

State Trajectory Input: [Already Selected: None] → [Remaining Candidates: 1. `birthPlace: London`, 2. `genre: pop`, . . .]
Task Constraint: Select ONE triple index that is MOST RELATED/CENTRAL to the entity.

The output of this stage is the sampled candidate sets $\mathcal{S}_R^t, \mathcal{S}_I^t, \mathcal{S}_C^t$ (valid unseen indices), and their merged representation $\tilde{\mathcal{A}}_t$, where

$$\tilde{\mathcal{A}}_t = \mathcal{S}_R^t \cup \mathcal{S}_I^t \cup \mathcal{S}_C^t, \quad \tilde{\mathcal{A}}_t \subseteq \mathcal{A}_t.$$

The merged set $\tilde{\mathcal{A}}_t$ is passed to the next stage.

Stage 2: Branch (Search Paths) The *Branch* stage shifts summarization from a linear selection process to multi-path tree exploration. It takes candidate index proposals $\tilde{\mathcal{A}}_t$ from the *Think* stage and instantiates competing summary trajectories. Given a parent node s_{t-1} and the proposed action set, the algorithm generates distinct successor nodes:

$$\{s_t^1, s_t^2, \ldots, s_t^{N_t}\}, \quad N_t \leq |\tilde{\mathcal{A}}_t|. \tag{6}$$

Here, N_t denotes the number of feasible triple transitions actually instantiated at depth t (after validity and duplicate filtering). Each successor appends exactly one candidate triple index from $\tilde{\mathcal{A}}_t$ to the parent state. The branch expansion at depth t yields a frontier of feasible partial summaries,

$$\mathcal{F}_t = \{s_t^1, s_t^2, \ldots, s_t^{N_t}\}.$$

Stage 3: Summarize (Scoring and Pruning)

Scoring. After generating alternative candidate paths $\mathcal{F}_t$, the framework applies a heuristic value function, denoted $V_{\mathcal{M}}(s)$, to convert the qualitative attributes of each partial summary state s to a scalar utility score. While both Stage 1 (Think) and this scoring component rely on the same three dimensions, they use them differently. Think (Stage 1) performs a local, per-triple decision: it evaluates each unselected triple with respect to the current state and processes the next best addition. Summarize (Stage 3), by contrast, is a global evaluator: it scores the entire partial summary after the triples have been assembled.

To achieve fine-grained calibration, the system provides the LLM with the state's accumulated triples and the entity description $\text{Desc}(e, \mathcal{T})$. The LLM acts as an analytical evaluator, scoring the trajectory across three independent semantic dimensions: $R(s)$ (Relatedness), $I(s)$ (Informativeness), and $C(s)$ (Coverage/diversity). The composite scalar utility of the state is then computed as a weighted linear combination:

$$V_{\mathcal{M}}(s) = w_r R(s) + w_i I(s) + w_c C(s) \tag{7}$$

where w_r, w_i, and w_c represent weights that determine the specific behavioral emphasis of the summarizer.

Prune (Beam Search Control). The final phase of each search cycle enforces search control through a dedicated *Prune* process. As the pool of open branches grows with depth, the framework applies a beam-search pruning operator to globally evaluate and rank all active candidates. At each search depth t, the operator merges the newly generated branches across the entire active frontier, evaluates them through the critic $V_{\mathcal{M}}(s)$, and discards low-utility branches according to a depth-dependent beam width:

$$S_t = \Phi_{B_t}(\mathcal{F}_t), \qquad \mathcal{F}_t = \bigcup_{s \in S_{t-1}} \text{Branch}\Big(s, \tilde{\mathcal{A}}_t(s)\Big), \tag{8}$$

where $B_t = 3$ for intermediate steps ($t < K$) and $B_t = 1$ at the final step ($t = K$). Thus, intermediate steps retain the top B_t highest-scoring partial trajectories, while the final step retains only the single best complete trajectory among the surviving candidates.

Following pruning, a conditional check in the breadth-first search loop determines the subsequent action. If $t < K$, the updated active frontier S_t is fed back into the Stage 1 *Think* policy for further expansion. If $t = K$, the loop terminates and returns the highest-scoring surviving complete trajectory as the final output entity summary S_e^*.

4 Experiment

In our experiments, we answer the following research questions:

RQ1: To what extent does a Tree of Thought strategy improve entity summarization quality?

RQ2: How do the individual components of ToT4ES—specifically the thought policy, branching strategy, and the three semantic dimensions (relatedness, informativeness, and coverage)—contribute to the overall summarization performance?

4.1 Datasets

To evaluate ToT4ES, we use two existing benchmarks: ESBM version 1.2 [17] and FACES [9]. Table 1 shows that ESBM comprises two subsets derived from DBpedia and LinkedMDB. The DBpedia subset represents general domains, such as agents, events, locations, species, and works, whereas LinkedMDB subset focuses on specific domains, such as films and people. Consistent with prior studies, we select 125 target entities from the DBpedia subset (comprising 2,721 entities and 4,436 relations) and 50 target entities from LinkedMDB subset (comprising 1,853 entities and 2,148 relations) for experimentation. Furthermore, FACES includes 50 target entities from DBpedia spanning diverse domains, such as persons, politicians, TV shows, songs, and universities. Overall, FACES contains 1,379 entities and 2,152 relations.

Table 1: Statistics for ESBM (DBpedia and LMDB) and FACES datasets.

Metric	DBpedia (ESBM)	LMDB (ESBM)	FACES
Entities ($\|\mathcal{V}\|$)	2 721	1 853	1 379
Relations ($\|\mathcal{E}\|$)	4 436	2 148	2 152
Target Entities	125	50	50

4.2 Evaluation Protocol

To assess the quality of the generated entity summaries (S_m), we use F-measure (F1-score), a standard metric in entity summarization [18]. This metric provides a balanced evaluation by calculating the harmonic mean of Precision (P) and Recall (R) with respect to a "ground truth" (S_h) set of triples selected by human experts. The F-measure is formally defined as follows:

$$\text{F-measure} = \frac{2.P.R}{P+R}, \text{ where } P = \frac{|S_m \cap S_h|}{|S_m|}, R = \frac{|S_m \cap S_h|}{|S_h|},$$

4.3 Baselines

In this study, we evaluate our approach against state-of-the-art unsupervised learning models.

- **RELIN** [3] adopts a random-surfer–based ranking scheme that integrates a centrality-oriented relatedness measure with statistical informativeness over description elements, producing concise entity summaries.
- **BAFREC** [16] balances frequency and rarity across entity properties, rather than merely selecting common concepts. It groups facts into distinct categories, scoring each via tailored metrics favoring rarity for type information (preferring specific concepts) and general popularity for remaining facts.
- **MPSUM** [30] extends probabilistic topic models (e.g., LDA) by incorporating predicate uniqueness and object importance for triple ranking, thereby generating concise, representative entity summaries.
- **KAFCA** [33] uses non-incremental Formal Concept Analysis for entity summarization. It ranks RDF triples via formal concept scores but requires complete summary recomputation upon arrival of new descriptions.
- **IRES** [20] explicitly exploits indirect relationships (e.g., second-hop neighbors) to derive compact, expressive entity descriptions. Grounded in graph-theoretical principles, it overcomes limitations of traditional first-hop-exclusive approaches.
- **Zero-shot LLM** [15]. This baseline adopts a direct single-pass prompting strategy, where the LLM is given the candidate RDF triples of an entity and asked to select exactly k triples that best satisfy three summarization criteria: *relatedness*, *informativeness*, and *coverage/diversity*.
- **Chain of Thought (CoT) LLM** [32]. This baseline uses the same candidate triples and the same summarization criteria as the Zero-shot LLM baseline,

namely *relatedness*, *informativeness*, and *coverage/diversity*. The difference is that, instead of directly selecting the summary triples in a single pass, the LLM is encouraged to perform explicit step-by-step reasoning over these criteria before producing the final subset of k triples.

Supervised methods such as ESA and ESLM are excluded from these comparisons because they require extensive human-labeled training data, whereas ToT4ES operates in a fully training-free setting. To ensure a fair comparison, ToT4ES is evaluated against standard LLM baselines using the same underlying LLM backbone and decoding parameters. Results for traditional unsupervised baselines (eg, RELIN, BAFREC) are reported directly from the benchmark publications [17] and their respective original papers (eg, IRES [20]), while all LLM-based baselines and ToT4ES variants were evaluated under identical experimental settings on the ESBM benchmark and FACES datasets.

4.4 Experimental Settings

LLM Settings. We evaluate ToT4ES using two open-source, instruction-tuned LLMs families: LLaMA[1] and Qwen[2]. These models are treated as frozen black boxes accessed only through direct prompting. During thought generation, we systematically investigate the use of temperatures within the range $[0.0, 0.9]$, while setting the temperature to 0.3 during the evaluation. In addition, during search, invalid indices (outside the valid range $[1, N_{triples}]$), out-of-range responses, and duplicate triple selection are automatically discarded.

Hyperparameter Settings. Given all triples of an entity, the Tree-of-Thought search grows candidate summaries for up to $K \in \{5, 10\}$ steps. At each step, the thinking policy consists of three task-specific LLM queries — one each for relatedness, informativeness, and diversity — each proposing up to $N' = 3$ candidate triple indices. Invalid indices (outside the valid range), duplicates, and out-of-range responses are automatically discarded, yielding approximately $N = 9$ total valid candidates per step (or fewer when invalid). The branching module applies beam pruning with a beam width of $B = 3$ at intermediate steps and $B = 1$ at the final step (K). The summarizing component then scores all candidate states using a weighted function (w_r, w_i, w_c) that aggregates three semantic dimensions: relatedness, informativeness, and coverage. To maintain a strictly training-free and dataset-agnostic setup, we adopt fixed global weights (i.e., $w_r = 0.4, w_i = 0.4, w_c = 0.2$) across all benchmark datasets without dataset-specific hyperparameter tuning.

Prompting Design for Summarize Component. The prompt for the Summarize component is designed to evaluate and score competing partial summaries generated during the search process. Following the candidate extraction in the Think

[1] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct
[2] Qwen/Qwen3-Coder-30B-A3B-Instruct

stage and the generation of alternative paths in the Branch phase, the Summarize module acts as a critic. It scores and ranks these partial summaries, enabling the framework to perform beam-search pruning and retain only the most promising summary trajectories. The structure of the prompt is as follows:

The template instructional prompt for summarize component

You are evaluating RDF triple summaries for the entity:

Entity: `{entity_label}`

Each candidate summary is a set of triples about this entity. You must rate each summary on:
- relatedness (0.0–1.0): how central and relevant the selected triples are to the entity. Emphasize triple_centrality and how core the predicates and values are to describing this entity.
- informativeness (0.0–1.0): how much important, non-trivial information the summary contains. Emphasize low freq_property and freq_value (rarer facts are more informative) and higher type_depth (more specific / deeper ontological types).
- coverage (0.0–1.0): how diverse and comprehensive the summary is across different aspects (types, roles, locations, time, etc.) while avoiding redundancy. Prefer summaries whose triples use different predicates/entity roles and whose values are not too similar.

5 Results and Discussion

5.1 Comparison of Performance Among State-of-the-Art Models (RQ1)

The results presented in Table 2 demonstrate that ToT4ES consistently outperforms both traditional baseline models (such as RELIN, IRES) and LLM baseline methods (Zero-shot, CoT) across all benchmarks, including DBpedia, LinkedMDB, and FACES datasets. A notable performance gap is observed between ToT4ES and LLM baseline methods. The Zero-shot LLM yields competitive results on the top $k = 5$, outperforming other approaches on FACES with an F-measure of 0.252. Incorporating reasoning steps through CoT LLM degrades performance below Zero-shot LLM across all datasets. Nevertheless, ToT4ES yields superior performance across nearly all benchmark settings. On DBpedia ($k = 5$) and FACES ($k = 10$), ToT4ES slightly outperforms the Zero-shot LLM baseline, while achieving significant improvement across DBpedia ($k = 10$) and the LinkedMDB ($k = 5, 10$), achieving F-measures of 0.573, 0.421, and 0.478, respectively. Our findings demonstrate that ToT4ES consistently satisfies length constraints (e.g., $k = 5, 10$), whereas LLM baselines frequently output incomplete or empty summaries. In particular, Chain-of-Thought (CoT) prompting fails to

Table 2: Performance evaluation (F-measure) of ToT4ES against baseline models on ESBM benchmark datasets (higher is better). Baseline scores of unsupervised methods for DBpedia and LinkedMDB are imported directly from canonical ESBM literature [17], whereas FACES scores are locally reproduced. Bold text indicates the best score among unsupervised/LLM training-free methods, and underlined scores represent the runner-up. (▲) denotes statistically significant improvement over the runner-up baseline ($p < 0.05$, paired two-tailed Wilcoxon signed-rank test).

	DBpedia		LinkedMDB		FACES	
Model	k=5	k=10	k=5	k=10	k=5	k=10
RELIN	0.242	0.455	0.203	0.258	0.114	0.254
MPSUM	0.314	0.512	0.272	0.423	0.152	0.286
BAPREC	0.335	0.503	0.360	0.402	0.131	0.258
KAFCA	0.314	0.509	0.244	0.397	0.122	0.298
IRES	0.327	<u>0.540</u>	<u>0.350</u>	<u>0.445</u>	0.128	0.265
Zero-shot LLM	<u>0.371</u>	0.531	0.306	0.353	**0.252**	<u>0.304</u>
Chain of Thought (CoT) LLM	0.246	0.504	0.277	0.339	<u>0.227</u>	0.191
TOT4ES (our)	**0.374**	**0.573▲**	**0.421▲**	**0.478▲**	0.212	**0.329**

generate valid summaries in up to 42% of evaluation cases on the FACES dataset ($k = 10$), severely degrading overall performance.

Compared to traditional baseline models, the superior improvement is mainly due to the tree-of-thought framework embedded in ToT4ES occurring in the LinkedMDB dataset, where it increases by approximately 20% at $k = 5$ compared to IRES. This architecture replaces traditional ranking-based selection with a multi-step deliberative search. By maintaining a beam of candidate states and evaluating them against three semantic dimensions—relatedness, informativeness, and coverage/diversity—the model captures the global context of an entity rather than merely selecting isolated triples. As a result, the generated summaries are more coherent and representative of an entity's diverse facets. The robustness of ToT4ES is further evidenced by its consistent performance across three distinct knowledge domains. On DBpedia, as a general-purpose knowledge base that often contains generic or redundant triples. ToT4ES achieves the highest F-measure (0.374 and 0.573) at $k = 5$ and $k = 10$, respectively. In LinkedMDB, this movie-centric dataset is characterized by dense relational interconnections, such as actor-director-film links. The substantial performance gap exceeds the nearest baseline (IRES) by over 0.07 at $k = 5$. Finally, on the FACES dataset, ToT4ES maintains a statistically significant lead across all unsupervised baselines for both summary lengths. By leveraging LLM reasoning within a structured search process, ToT4ES maintains high summary quality even under strict size constraints, positioning ToT4ES as the state-of-the-art method for unsupervised entity summarization.

Table 3: Impact of LLM temperature settings on ToT4ES performance, evaluated based on F-measure on benchmarks (higher is better).

	DBpedia		LinkedMDB		FACES	
Model	k=5	k=10	k=5	k=10	k=5	k=10
ToT4ES ($Temperature = 0.0$)	0.335	0.527	0.413	0.463	0.202	0.311
ToT4ES ($Temperature = 0.3$)	0.339	0.522	0.412	**0.478**	0.203	0.312
ToT4ES ($Temperature = 0.5$)	0.336	0.512	0.396	0.474	0.196	0.324
ToT4ES ($Temperature = 0.7$)	0.337	0.509	0.413	0.467	0.198	0.317
ToT4ES ($Temperature = 0.8$)	**0.374**	**0.573**	**0.421**	**0.478**	**0.212**	**0.329**
ToT4ES ($Temperature = 0.9$)	0.330	0.518	0.418	0.467	0.205	0.313

5.2 Hyperparameter Sensitivity Analysis

Effect of LLM Decoding Temperature A systematic sensitivity analysis of the ToT4ES framework was conducted using `Qwen3-Coder-30B-A3B-Instruct` backbone across a temperature range of $[0.0, 0.9]$. The results, summarized in Table 3, show that ToT4ES is robust to variations in decoding stochasticity. For general entities in DBpedia, a temperature of 0.8 achieved the peak F-measure of 0.374 and 0.573 for $k = 5$ and $k = 10$, respectively. This setting also benefited LinkedMDB and FACES by encouraging exploration; specifically, $Temperature = 0.8$ yielded the highest F-measure for both LinkedMDB (0.421) and FACES (0.212) at $k = 5$, as well as LinkedMDB (0.478) and FACES (0.329) at $k = 10$

These findings support the intended design of the Tree of Thought architecture. The framework functions as a synergistic mechanism for exploration and pruning. Higher temperatures encourage the "Thinking" stage to propose non-obvious or highly specific candidate triples, while the "Summarizing" (Critic) component effectively prunes low-utility noise. This approach enables the system to leverage LLM creativity for discovery while maintaining the factual rigor necessary for concise entity summarization.

5.3 Ablation Study (RQ2)

Comparison ToT4ES using Different LLMs Table 4 presents the performance of ToT4ES when evaluated with different LLM backbones. The results demonstrate that while the framework is model-agnostic, its selection fidelity scales positively with the model's reasoning capacity and parameter size. The larger `Qwen3-Coder-30B-A3B-Instruct` variant consistently outperforms `Llama-3.1-8B-Instruct` across all three datasets and both summary lengths ($k = 5, 10$). The performance advantage is most pronounced on DBpedia and LinkedMDB, where switching to the 30B model yields absolute$F$-measure increases of up to 0.094 (on DBpedia at $k = 10$) and 0.078 (on LinkedMDB at $k = 5$). These gains indicate that stronger reasoning capabilities in the thought-

Table 4: Average F-measure across benchmarks (higher is better).

	DBpedia		LinkedMDB		FACES	
Model	k=5	k=10	k=5	k=10	k=5	k=10
Llama-3.1-8B-Instruct	0.307	0.479	0.343	0.427	0.202	0.314
Qwen3-Coder-30B-A3B-Instruct	**0.374**	**0.573**	**0.421**	**0.478**	**0.212**	**0.329**

generation and heuristic evaluation steps enable the search process to prune suboptimal candidate branches more effectively.

Comparison ToT4ES Component Contribution We conducted an extensive ablation study to evaluate the individual contributions of the modular components within the ToT4ES framework, comparing the full pipeline against seven constrained scenarios: (i) No Branching (Greedy, $B = 1$), (ii) Relatedness (R) Only, (iii) Informativeness (I) Only, (iv) Coverage (C) Only, (v) Relatedness & Informativeness (R & I), (vi) Relatedness & Coverage (R & C), and (vii) ToT4ES without LLM scoring on Stage 3 (Summarize). The results in Table 5 demonstrate that each evaluation dimension provides an indispensable heuristic signal, as disabling any single objective leads to a consistent performance drop across all datasets. In isolation, omitting Coverage ($C(s)$) causes the model to select semantically redundant triples, whereas removing Informativeness ($I(s)$) allows generic, non-discriminative facts to pass through.

Several critical insights regarding the framework's architecture: The "ToT4ES (No Branch)" variant represents a purely greedy approach where the beam width is set to $B = 1$, retaining only the single highest-scoring state at each step. While this greedy baseline performs surprisingly well on DBpedia (0.340 at $k = 5$), it is generally inferior on FACES dataset.

On DBpedia, the Full Pipeline achieves optimal performance at $k = 10$ (0.573). Under strict budget constraints at $k = 5$, C Only (0.359) outperforms both R Only (0.341) and I Only (0.333), while eliminating LLM-based scoring entirely results in the lowest performance (0.258). These results suggest that, for general-purpose entities in DBpedia, the Coverage/Diversity dimension effectively captures the necessary mix of high-salience taxonomic facts when the size constraint is limited. In contrast, Coverage alone demonstrates significant limitations on LinkedMDB, achieving only 0.289 at $k = 5$, which demonstrates that the integration of all three dimensions is essential for balanced performance across diverse knowledge domains.

The most notable result appears in the FACES dataset at $k = 10$, where I-only (0.329) surpasses the Full Pipeline (0.318). This counterintuitive result—where a single dimension outperforms its combination—validates the key observation: the optimal utility function is highly sensitive to the topological and thematic structure of the KG. On sparse, high-dimensional entity graphs like FACES, Informativeness alone proves sufficient and even advantageous, suggesting that for entities with diverse but focused facets, specificity matters more than breadth.

Table 5: Average F-measure across benchmarks (higher is better).

	DBpedia		LinkedMDB		FACES	
Model	k=5	k=10	k=5	k=10	k=5	k=10
ToT4ES (Full Pipeline)	**0.374**	**0.573**	**0.421**	0.478	**0.212**	0.318
ToT4ES (No Branch)	0.340	0.511	0.418	**0.484**	0.206	0.315
ToT4ES (R Only)	0.341	0.518	0.406	0.467	0.207	0.306
ToT4ES (I Only)	0.333	0.514	0.413	0.469	0.194	**0.329**
ToT4ES (C Only)	0.359	0.539	0.289	0.407	0.195	0.317
ToT4ES (R & I)	0.327	0.510	0.302	0.416	0.186	0.320
ToT4ES (R & C)	0.337	0.518	0.320	0.409	0.197	0.313
ToT4ES (Without LLM Scoring)	0.258	0.422	0.246	0.380	0.156	0.277

The pairwise combined metrics (R & I, R & C) generally exhibit suboptimal performance on DBpedia and LinkedMDB, frequently underperforming compared to single-objective evaluations due to heuristic interference when the balancing third dimension is omitted. Notably, R & I attains a score of 0.320 on FACES at $k = 10$, marginally surpassing the Full Pipeline (0.318). These results indicate that, in high-dimensional graphs, two well-aligned dimensions can capture essential structural information without the dilution caused by competing diversity constraints. Overall, the Full Pipeline consistently outperforms alternative variants across nearly all settings, demonstrating that multi-dimensional heuristic synergy is essential for training-free, generalizable entity summarization.

Limitation. ToT4ES operates more slowly than traditional unsupervised methods because it repeatedly invokes an LLM during the search process. The computational cost scales with the summary budget k. In the task-decomposed variant, each step may require three candidate-generation calls, in addition to multiple evaluations, resulting in significantly higher costs than lightweight baselines.

6 Conclusion and Future Work

We introduce ToT4ES, a framework that formulates entity summarization as a deliberative search process powered by LLMs. By moving beyond traditional rank-and-pick extractive methods toward a Tree of Thought architecture, this approach enables more nuanced reasoning about an entity's semantic identity. The Think, Branch, and Summarize modules allow the system to explicitly optimize for relatedness, informativeness, and coverage. Experimental results across DBpedia, LinkedMDB, and FACES demonstrate strong performance, with particularly notable gains on LinkedMDB, where broader context appears to be important. These findings indicate that structured LLM reasoning offers a viable training-free alternative, addressing the rigidity of isolated unsupervised heuristics while offering greater domain transferability than data-intensive supervised models.

Future work will focus on reducing LLM inference cost through adaptive branching and fewer evaluation calls, improving robustness with dataset-aware or learned weighting of semantic dimensions, and extending the framework to additional benchmarks and domains. Another promising direction is the incorporation of automatic prompt optimization and human evaluation to better study summary quality, factuality, and diversity.

Declaration on Generative AI

Generative AI tools, such as Gemini and Grammarly, were used to check and improve English grammar and refine the readability of selected paragraphs of the manuscript. All AI-generated text is reviewed, edited, and approved by the authors, who take all responsibility for the final text, and confirm that the AI system is credited as a co-author, in accordance with the conference's generative AI policy.

Acknowledgment

This work has been supported by the German Federal Ministry of Research, Technology and Space (BMFTR) within the project KI-Akademie OWL under the grant no. 16IS24057B, by the Federal Ministry for Economic Affairs and Energy (BMWE) on the basis of a decision by the German Bundestag within the project ikDS under the grant no. KK5175206LO4, and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) within the project TRR 318/3 2026 under the grant no. 438445824.

Disclosure of Interests. The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Abu-Salih, B., Al-Qurishi, M., Alweshah, M., Al-Smadi, M., Alfayez, R., Saadeh, H.: Healthcare knowledge graph construction: A systematic review of the state-of-the-art, open issues, and opportunities. J. Big Data **10**(1), 81 (2023). https://doi.org/10.1186/S40537-023-00774-9, https://doi.org/10.1186/s40537-023-00774-9
2. Bengio, Y., Ducharme, R., Vincent, P., Janvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. **3**, 1137–1155 (2003), https://jmlr.org/papers/v3/bengio03a.html
3. Cheng, G., Tran, T., Qu, Y.: RELIN: relatedness and informativeness-based centrality for entity summarization. In: The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I. Lecture Notes in Computer Science, vol. 7031, pp. 114–129. Springer (2011). https://doi.org/10.1007/978-3-642-25073-6_8, https://link.springer.com/chapter/10.1007/978-3-642-25073-6_8

4. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers). pp. 4171–4186. Association for Computational Linguistics (2019). https://doi.org/10.18653/V1/N19-1423, https://doi.org/10.18653/v1/n19-1423
5. Ermilov, T., Moussallem, D., Usbeck, R., Ngomo, A.N.: GENESIS: a generic RDF data access interface. In: Sheth, A.P., Ngonga, A., Wang, Y., Chang, E., Slezak, D., Franczyk, B., Alt, R., Tao, X., Unland, R. (eds.) Proceedings of the International Conference on Web Intelligence, Leipzig, Germany, August 23-26, 2017. pp. 125–131. ACM (2017). https://doi.org/10.1145/3106426.3106514, https://doi.org/10.1145/3106426.3106514
6. Firmansyah, A.F., Moussallem, D., Ngomo, A.N.: GATES: using graph attention networks for entity summarization. In: Gentile, A.L., Gonçalves, R. (eds.) K-CAP '21: Knowledge Capture Conference, Virtual Event, USA, December 2-3, 2021. pp. 73–80. ACM (2021). https://doi.org/10.1145/3460210.3493574, https://doi.org/10.1145/3460210.3493574
7. Firmansyah, A.F., Moussallem, D., Ngomo, A.N.: ESLM: improving entity summarization by leveraging language models. In: The Semantic Web - 21st International Conference, ESWC 2024, Hersonissos, Crete, Greece, May 26-30, 2024, Proceedings, Part I. Lecture Notes in Computer Science, vol. 14664, pp. 162–179. Springer (2024). https://doi.org/10.1007/978-3-031-60626-7_9, https://link.springer.com/chapter/10.1007/978-3-031-60626-7_9
8. Firmansyah, A.F., Zahera, H.M., Sherif, M.A., Moussallem, D., Ngomo, A.N.: ANTS: abstractive entity summarization in knowledge graphs. In: The Semantic Web - 22nd European Semantic Web Conference, ESWC 2025, Portoroz, Slovenia, June 1-5, 2025, Proceedings, Part I. Lecture Notes in Computer Science, vol. 15718, pp. 133–151. Springer (2025). https://doi.org/10.1007/978-3-031-94575-5_8, https://link.springer.com/chapter/10.1007/978-3-031-94575-5_8
9. Gunaratna, K., Thirunarayan, K., Sheth, A.P.: FACES: diversity-aware entity summarization using incremental hierarchical conceptual clustering. In: Bonet, B., Koenig, S. (eds.) Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA. pp. 116–122. AAAI Press (2015). https://doi.org/10.1609/AAAI.V29I1.9180, https://dl.acm.org/doi/10.5555/2887007.2887024
10. Hasibi, F., Balog, K., Bratsberg, S.E.: Dynamic factual summaries for entity cards. In: Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A.P., White, R.W. (eds.) Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017. pp. 773–782. ACM (2017). https://doi.org/10.1145/3077136.3080810, https://doi.org/10.1145/3077136.3080810
11. Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G.D., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., Ngomo, A.C.N., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., Zimmermann, A.: Knowledge graphs. ACM Comput. Surv. **54**(4) (Jul 2021). https://doi.org/10.1145/3447772, https://doi.org/10.1145/3447772
12. Ilkou, E., Jiménez-Ruiz, E.: Towards a knowledge graph for teaching knowledge graphs. In: Etcheverry, L., Garcia, V.L., Osborne, F., Pernisch, R. (eds.) Proceedings of the ISWC 2024 Posters, Demos and Industry Tracks: From Novel Ideas to

Industrial Practice co-located with 23nd International Semantic Web Conference (ISWC 2024), Hanover, Maryland, USA, November 11-15, 2024. CEUR Workshop Proceedings, CEUR-WS.org (2024), https://ceur-ws.org/Vol-3828/paper10.pdf
13. Jeyaraman, B.P., Dai, B.T., Fang, Y.: A comprehensive review of financial knowledge graphs. World Sci. Annu. Rev. Artif. Intell. **3**, 2530001:1–2530001:14 (2025). https://doi.org/10.1142/S2811032325300014, https://doi.org/10.1142/S2811032325300014
14. Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., Mikolov, T.: Fasttext.zip: Compressing text classification models. CoRR **abs/1612.03651** (2016), http://arxiv.org/abs/1612.03651
15. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are zero-shot reasoners. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022 (2022), http://papers.nips.cc/paper_files/paper/2022/hash/8bb0d291acd4acf06ef112099c16f326-Abstract-Conference.html
16. Kroll, H., Nagel, D., Balke, W.T.: Bafrec: Balancing frequency and rarity for entity characterization in linked open data. In: Proc. 1st Int. Workshop Entity REtrieval (2018), https://www.hkroll.de/pdf/EYRE2018-kroll-BAFREC.pdf
17. Liu, Q., Cheng, G., Gunaratna, K., Qu, Y.: ESBM: an entity summarization benchmark. In: Harth, A., Kirrane, S., Ngomo, A.N., Paulheim, H., Rula, A., Gentile, A.L., Haase, P., Cochez, M. (eds.) The Semantic Web - 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31-June 4, 2020, Proceedings. pp. 548–564. Lecture Notes in Computer Science, Springer (2020). https://doi.org/10.1007/978-3-030-49461-2_32, https://doi.org/10.1007/978-3-030-49461-2_32
18. Liu, Q., Cheng, G., Gunaratna, K., Qu, Y.: Entity summarization: State of the art and future challenges. Journal of Web Semantics **69**, 100647 (2021). https://doi.org/https://doi.org/10.1016/j.websem.2021.100647, https://www.sciencedirect.com/science/article/pii/S1570826821000226
19. Liu, Q., Cheng, G., Qu, Y.: Deeplens: Deep learning for entity summarization. In: Alam, M., Buscaldi, D., Cochez, M., Osborne, F., Recupero, D.R., Sack, H. (eds.) Proceedings of the Workshop on Deep Learning for Knowledge Graphs (DL4KG2020) co-located with the 17th Extended Semantic Web Conference 2020 (ESWC 2020), Heraklion, Greece, June 02, 2020 - moved online. CEUR Workshop Proceedings, vol. 2635. CEUR-WS.org (2020), https://ceur-ws.org/Vol-2635/paper2.pdf
20. Moradan, A., Sorkhpar, M., Miyauchi, A., Mottin, D., Assent, I.: Untapping the power of indirect relationships in entity summarization. In: Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining. p. 820–828. WSDM '25, Association for Computing Machinery, New York, NY, USA (2025). https://doi.org/10.1145/3701551.3703566, https://doi.org/10.1145/3701551.3703566
21. Peng, C., Xia, F., Naseriparsa, M., Osborne, F.: Knowledge graphs: Opportunities and challenges. Artif. Intell. Rev. **56**(11), 13071–13102 (2023). https://doi.org/10.1007/S10462-023-10465-9, https://doi.org/10.1007/s10462-023-10465-9
22. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Moschitti, A., Pang, B., Daelemans, W. (eds.) Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL. pp. 1532–1543. ACL (2014). https://doi.org/10.3115/V1/D14-1162, https://doi.org/10.3115/v1/d14-1162

23. Pouriyeh, S., Allahyari, M., Kochut, K., Cheng, G., Arabnia, H.R.: ES-LDA: Entity summarization using knowledge-based topic modeling. In: Kondrak, G., Watanabe, T. (eds.) Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 316–325. Asian Federation of Natural Language Processing, Taipei, Taiwan (Nov 2017), https://aclanthology.org/I17-1032/
24. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**, 140:1–140:67 (2020), https://jmlr.org/papers/v21/20-074.html
25. Sun, Y., Wang, S., Li, Y., Feng, S., Tian, H., Wu, H., Wang, H.: ERNIE 2.0: A continual pre-training framework for language understanding. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020. pp. 8968–8975. AAAI Press (2020). https://doi.org/10.1609/AAAI.V34I05.6428, https://doi.org/10.1609/aaai.v34i05.6428
26. Thalhammer, A., Lasierra, N., Rettinger, A.: Linksum: Using link analysis to summarize entity data. In: Web Engineering - 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings. Lecture Notes in Computer Science, vol. 9671, pp. 244–261. Springer (2016). https://doi.org/10.1007/978-3-319-38791-8_14, https://link.springer.com/chapter/10.1007/978-3-319-38791-8_14
27. Trouli, G.E., Vassiliou, G., Giannakos, I., Papadakis, N., Kondylakis, H.: Entitysum: Entity summarization using centrality measures. In: Curry, E., Presutti, V., McCrae, J., Alam, M., Colpaert, P., Xavier Parreira, J., Collarana, D., Sabou, M., Harth, A., Lisena, P. (eds.) The Semantic Web: ESWC 2025 Satellite Events. pp. 29–33. Springer Nature Switzerland, Cham (2026), https://link.springer.com/chapter/10.1007/978-3-031-99554-5_6
28. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. CoRR **abs/1710.10903** (2017), http://arxiv.org/abs/1710.10903
29. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: Brodley, C.E., Stone, P. (eds.) Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada. pp. 1112–1119. AAAI Press (2014). https://doi.org/10.1609/AAAI.V28I1.8870, https://doi.org/10.1609/aaai.v28i1.8870
30. Wei, D., Gao, S., Liu, Y., Liu, Z., Hang, L.: Mpsum: entity summarization with predicate-based matching. arXiv preprint arXiv:2005.11992 (2020), https://arxiv.org/abs/2005.11992
31. Wei, D., Liu, Y., Zhu, F., Zang, L., Zhou, W., Han, J., Hu, S.: ESA: entity summarization with attention. In: Cheng, G., Gunaratna, K., Wang, J. (eds.) Proceedings of the 2nd International Workshop on EntitY REtrieval co-located with 28th ACM International Conference on Information and Knowledge Management (CIKM 2019), Beijing, China, November 3, 2019. CEUR Workshop Proceedings, vol. 2446, pp. 40–44. CEUR-WS.org (2019), https://ceur-ws.org/Vol-2446/paper6.pdf
32. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E.H., Le, Q.V., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A.

(eds.) Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022 (2022), http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html

33. Yang, E., Hao, F., Yang, Y., De Maio, C., Nasridinov, A., Min, G., Yang, L.T.: Incremental entity summarization with formal concept analysis. IEEE Transactions on Services Computing **15**(6), 3289–3303 (2022). https://doi.org/10.1109/TSC.2021.3090276, https://ieeexplore.ieee.org/document/9459533