BART-Dijkstra-SA:

A Scalable Workflow for Semantic Summarization and Topic Structuring of Large-Scale Text Data

Author: Chen-Hong

Affiliation: National Chung Cheng University

Date: April 26, 2025

Abstract

As the global volume of information continues to grow exponentially, the management, comprehension, and generation of large-scale text data have become core challenges in the fields of Natural Language Processing (NLP) and Big Data analytics. Addressing the limitations of existing generative models in thematic focus and semantic consistency, this study proposes the BART-Dijkstra-SA (BDS) workflow as a systematic solution for large-scale text topic modeling and summarization. The BDS workflow integrates simulated annealing to optimize the number of topics, applies Latent Dirichlet Allocation (LDA) and TF-IDF vectorization to construct semantic graphs, and uses the Leiden algorithm to ensure community connectivity and modularity stability. Subsequently, Dijkstra's shortest path algorithm is employed to extract semantically coherent keyword sequences, which serve as structural prompts to guide the BART model in generating focused and semantically consistent text summaries and titles. Under conditions of increasing data scale, the proposed method demonstrates superior scalability and generation stability. The BDS workflow is broadly applicable to large-scale topic modeling, text summarization, and knowledge management, offering a new generation of systematic solutions for natural language understanding and generation technologies.

1. Introduction

With the exponential growth of global information, the management, understanding, and generation of large-scale text data have become core challenges in natural language processing (NLP) and big data analytics (Reuters Institute for the Study of, 2023; Silverman, 2016). In application scenarios such as misinformation analysis, news monitoring, and topic modeling, text data exhibit complex and intertwined semantic structures, imposing stricter requirements on generative models regarding thematic focus, semantic consistency, and structural grasp.

Mainstream generative pre-trained models, such as BART (Lewis et al., 2019), demonstrate robust language comprehension and generation capabilities. However, due to the computational complexity of the self-attention mechanism ($O(n^2)$), they are prone to semantic drift and thematic defocusing when handling million-scale samples or multi-themed long texts. Additionally, semantic graph construction techniques, such as TF-IDF vectorization and cosine similarity calculations, also exhibit $O(k^2)$ computational complexity during keyword set expansions, creating performance bottlenecks. Traditional community clustering methods, like the Louvain algorithm (Blondel et al., 2008), although capable of quickly approximating modularity maximization, face issues of community fragmentation and clustering distortion as sample sizes grow (Traag et al., 2019). Their complexity, $O(n \log n)$, is inadequate for supporting ultra-large-scale semantic network processing demands.

In big data environments, data structure design, algorithmic time complexity control, and workflow automation have become critical factors determining the scalability of text generation and comprehension systems. Traditional methods relying on manual parameter tuning, brute-force searches, and unguided generation become inefficient and degrade in output quality as the number of samples (n) rapidly increases, causing resource bottlenecks and performance deterioration.

To address these challenges, this study proposes the BART-Dijkstra-SA (BDS) workflow, a systematic solution for processing large-scale text data. This workflow integrates structural modeling, heuristic optimization, and generation guidance through three core technologies:

1. Topic Number Optimization (Simulated Annealing, SA): Incorporating the Simulated Annealing algorithm (Kirkpatrick et al., 1983) with Perplexity as the objective function to automatically explore the topic number space. Compared to traditional grid searches ($O(k \cdot n)$), simulated annealing achieves optimal topic number selection at an approximate complexity of $O(\log k \cdot n)$, significantly reducing computational burdens in large-scale data.
2. Topic Modeling and Semantic Graph Construction (LDA + TF-IDF + Cosine Similarity): Performing LDA topic modeling based on the optimal topic number, constructing a keyword semantic graph via TF-IDF vectorization and cosine similarity calculations, laying the foundation for subsequent structural analysis.
3. Community Clustering (Leiden Algorithm): Utilizing the Leiden algorithm for community clustering ensures connectivity and modular stability within semantic graphs, maintaining a time complexity of $O(m)$, where m represents the number of edges.
4. Shortest Semantic Path Search (Dijkstra Algorithm): Employing the Dijkstra shortest path algorithm to extract semantically coherent keyword sequences for each topic community. Given the small community sizes, an array-based implementation efficiently performs local path searches at a complexity of $O(V^2)$.
5. Structure-guided Generation (BART): Using the shortest semantic paths as structural prompts, guiding the BART model to generate topic-focused and semantically consistent text summaries and topic titles.

Overall, the BART-Dijkstra-SA workflow balances deep structural modeling, controllable computational complexity, and stable generation quality. Its scalability becomes increasingly advantageous as sample sizes expand, showcasing substantial potential for broad applications in topic modeling, text summarization, and knowledge management. This method represents a next-generation systematic solution for NLP and text analytics.

2. Related Work

Text Generation and Semantic Modeling Techniques

Text generation techniques continue to evolve in the field of natural language processing (NLP). The development of generative pre-trained models, notably BART (Bidirectional and Auto-Regressive Transformers) (Lewis et al., 2019), significantly enhances machine capabilities in language understanding and generation. BART integrates both encoder and decoder architectures through a denoising autoencoder approach, effectively repairing damaged text and performing high-quality generation tasks. Its training methodology simulates masking, shuffling, and deleting sentence segments, thus reinforcing semantic structural understanding and becoming critical in tasks such as summarization, translation, and inference.

On the other hand, topic modeling serves as a crucial technique for comprehending textual thematic structures, with the Latent Dirichlet Allocation (LDA) model (Blei et al., 2003) widely applied in large-scale text analysis. LDA assumes documents consist of several latent topics, each defined by word distributions, automatically revealing hidden structures within texts. However, LDA is sensitive to topic number configurations and tends to fall into local optima in complex topic spaces (Pathik & Shukla, 2020).

For semantic graph construction, TF-IDF (Term Frequency-Inverse Document Frequency) and cosine similarity are commonly used measures for word similarity, effectively quantifying semantic proximity between texts (Zhang et al., 2011). Such methods support constructing texts or topic keywords into graph structures, forming the basis for subsequent semantic path search and structural analysis.

Nevertheless, existing methods still face efficiency and semantic focusing challenges as the number of samples and topics dramatically increase. This study addresses these limitations by proposing a structurally guided text generation workflow to enhance thematic consistency and summary quality.

Semantic Structure Construction and Community Detection Techniques

Community detection within semantic graph structures is critical for understanding inter-topic relationships and text structure. Traditional methods, such as the Louvain algorithm (Blondel et al., 2008), maximize modularity and perform fast clustering with a near O(n log n) complexity. However, further research indicates the Louvain method sometimes exhibits issues such as insufficient community connectivity and elevated modularity errors, leading to distorted topic structures (Traag et al., 2019).

To overcome these issues, the Leiden algorithm (Traag et al., 2019) optimizes the community clustering process, ensuring node connectivity within each community and enhancing modularity stability and computational efficiency. The Leiden algorithm

maintains a complexity of O(m), where m represents edge count, making it highly suitable for managing the expanding scale of semantic graphs in big data environments.

Additionally, graph theory's Dijkstra algorithm (Dijkstra, 1959) is widely employed to extract the most semantically coherent association paths between texts or topics. Dijkstra efficiently finds single-source shortest paths in graphs with non-negative weights, with an array-based implementation complexity of $O(V^2)$, where V denotes node count. In moderate keyword number scenarios, it effectively searches for the shortest semantic connections to guide focused text generation.

Given the strengths and limitations of current community clustering and shortest path search techniques, this study particularly emphasizes clustering accuracy, path search efficiency, and semantic consistency integration as critical considerations in the method design.

Large-Scale Text Processing and Complexity Control

In the context of big data, exponential growth in text data poses significant resource bottlenecks and efficiency issues for traditional brute-force computations or non-optimized workflows (Kannan et al., 2016). Hence, controlling algorithmic time complexity and automating workflows become key factors determining system scalability.

In topic modeling, traditional hyperparameter optimization methods like random initialization or grid searches typically have a complexity of $O(k \cdot n)$, highly time-consuming for large-scale data. To address this, heuristic optimization methods such as Simulated Annealing (SA) (Kirkpatrick et al., 1983) have proven capable of automating topic number searches with near $O(\log k \cdot n)$ efficiency, avoiding local optima and reducing computational load (Pathik & Shukla, 2020).

Moreover, effectively selecting representative prompt words after semantic graph construction and community clustering is crucial for improving system efficiency and semantic quality. The shortest path prompt word strategy controls computational resource consumption while enhancing semantic focus and structural consistency, achieving precise and efficient text generation goals.

In summary, the method design in this study prioritizes high scalability, low complexity, and semantic stability. By integrating topic optimization, community clustering, shortest path search, and structurally guided generation techniques, it proposes the BART-Dijkstra-SA (BDS) workflow as a systematic solution addressing the challenges in large-scale text comprehension and generation.

3. Methodology

Overview

To address challenges in topic modeling, summarization, and structural comprehension of large-scale text data, this study proposes the BART-Dijkstra-SA (BDS) workflow. This workflow integrates structural modeling, heuristic optimization, and generation guidance, balancing semantic focus, computational efficiency, and scalability. The overall workflow is as follows:

1. Automatically optimize the number of topics using Simulated Annealing (SA), reducing the computational burden associated with manual tuning and brute-force searches.
2. Perform topic modeling using Latent Dirichlet Allocation (LDA), constructing semantic graphs through TF-IDF vectorization and cosine similarity calculations.
3. Employ the Leiden algorithm for community detection, ensuring connectivity and modular stability among topic clusters.
4. Utilize the Dijkstra shortest path algorithm to extract semantically coherent keyword sequences from each topic cluster.
5. Use these keywords as structural prompts to guide the BART model in generating focused, semantically consistent summaries and topic titles.

This workflow design aims to balance system scalability in big data environments with semantic quality requirements for natural language generation tasks.

Topic Number Optimization (Simulated Annealing)

The selection of the number of topics (k) significantly impacts the performance of topic modeling. Traditional grid search methods iterate through multiple candidate parameters, leading to high computational complexity of $O(k \cdot n)$ and considerable burdens on large datasets (Hasan et al., 2020).

To address this, the study adopts Simulated Annealing (SA) (Kirkpatrick et al., 1983) as a mechanism for exploring optimal topic numbers. SA employs a controlled stochastic search mechanism, simulating physical annealing processes, greatly reducing search iterations and effectively avoiding local optima. The LDA perplexity serves as the energy function, converging to a near-global optimal topic number through temperature reduction and neighborhood search strategies. This approach reduces computational complexity to approximately $O(\log k \cdot n)$, significantly enhancing feasibility in large-scale data processing environments.

Topic Modeling and Semantic Graph Construction (LDA + TF-IDF + Cosine Similarity)

After optimizing the topic number, the LDA model (Blei et al., 2003) is utilized for topic modeling. LDA assumes each document comprises multiple topics described by word probability distributions, effectively capturing latent semantic structures within texts.

To construct a semantic graph of topic keywords, this study uses TF-IDF vectorization, followed by cosine similarity to calculate word similarity. Keywords serve as nodes, with similarity scores as edge weights, forming a semantic graph. This graph structure provides a foundation for subsequent community clustering and shortest semantic path searches.

Community Clustering (Leiden Algorithm)

To understand structural relationships between topics, community clustering is applied to the semantic graph. Traditional methods, such as the Louvain algorithm (Blondel et al., 2008), quickly approximate modularity maximization but often exhibit issues like community fragmentation and instability (Traag et al., 2019).

Therefore, this study employs the Leiden algorithm (Traag et al., 2019), which performs detailed optimizations and connectivity checks after each partitioning, ensuring full node connectivity within each community and significantly improving modular stability. With a time complexity of O(m), the Leiden algorithm effectively manages large-scale semantic network structures, supporting downstream shortest path keyword selection tasks.

Shortest Semantic Path Search (Dijkstra Algorithm)

For each topic cluster obtained from community clustering, this study uses the Dijkstra shortest path algorithm (Dijkstra, 1959) to search for the shortest, most coherent semantic paths among keywords in the semantic graph. This method effectively extracts representative keyword sequences to guide generation.

Given the limited number of keywords within communities (typically around a dozen), an array-based implementation of the Dijkstra algorithm is adopted, offering efficient performance with a time complexity of $O(V^2)$ within localized scopes.

Structure-Guided Generation (BART Model)

Finally, this study combines the shortest semantic path-derived keywords with the original text summary as structural prompts, inputting these into the BART model (Lewis et al., 2019).

By explicitly guiding the generation toward focused topics, this workflow significantly reduces semantic drift and topic defocusing issues common to BART without structural guidance, enhancing the focus, consistency, and readability of generated summaries.

Overall, the BART-Dijkstra-SA workflow balances semantic depth and computational efficiency, demonstrating substantial scalability and application potential in large-scale text processing scenarios.

References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, *3*, 993-1022.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, *1*, 269-271. https://doi.org/10.1007/BF01386390

Hasan, M., Rahman, A., Razaul, M., Khan, M., & Islam, M. (2020). Normalized Approach to Find Optimal Number of Topics in Latent Dirichlet Allocation (LDA). *International Journal of Advanced Computer Science and Applications*.

Kannan, S., Karuppusamy, S., Nedunchezhian, A., Venkateshan, P., Wang, P., Bojja, N., & Kejariwal, A. (2016). Big Data Analytics for Social Media. In *Big Data* (pp. 63-94). Academic Press. https://doi.org/10.1016/B978-0-12-805394-2.00003-9

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, *220*(4598), 671-680. https://doi.org/10.1126/science.220.4598.671

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *arXiv preprint arXiv:1910.13461*. https://arxiv.org/abs/1910.13461

Pathik, N., & Shukla, P. (2020). Simulated Annealing Based Algorithm for Tuning LDA Hyper Parameters. In *Soft Computing: Theories and Applications*. Springer. https://doi.org/10.1007/978-981-15-4032-5_47

Reuters Institute for the Study of, J. (2023). Digital News Report 2023. https://reutersinstitute.politics.ox.ac.uk/digital-news-report/2023

Silverman, C. (2016). *This Analysis Shows How Viral Fake Election News Stories Outperformed Real News on Facebook*. https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook

Traag, V. A., Waltman, L., & van Eck, N. J. (2019). From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, *9*, 5233. https://doi.org/10.1038/s41598-019-41695-z

Zhang, W., Yoshida, T., & Tang, X. (2011). A comparative study of TF*IDF, LSI and multi-words for text classification. *Expert Systems with Applications*, *38*, 2758-2765.

https://doi.org/10.1016/j.eswa.2010.08.066