Tailored Network Design with AI

Tanmay Pant
Dept. Name of Organization (of Aff.)
Name of Organization (of Aff.)
City, Country
tanmay.pant@mitaoe.ac.in

Soham Dabhade
Dept. Name of Organization (of Aff.)
Name of Organization (of Aff.)
City, Country
soham.dabhade@mitaoe.ac.in

Jyoti Choudhary
Dept. Name of Organization (of Aff.)
Name of Organization (of Aff.)
City, Country
jyoti.choudhary@mitaoe.ac.in

***Abstract*—NetOptiAI revolutionizes network infrastructure design by harnessing the power of artificial intelligence. This innovative system empowers users to define their network priorities, from server counts to performance metrics, enabling the AI to craft optimal network topologies. NetOptiAI delivers personalized solutions that balance user preferences and technical constraints, ensuring efficient, robust, and tailored network configurations.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

NetOptiAI addresses the growing challenges in modern network design. As networks become more complex, traditional methods struggle to meet diverse user needs and technical demands. Manual design processes are often slow and can't handle scala-bility or different user needs effectively. NetOptiAI aims to change this by allowing users to clearly define their network priorities and constraints.The system creates custom network designs that balance user preferences with technical requirements.

A. Project solution

NetOptiAI transforms network design to include user-defined priorities. Users specify their needs, like the number of servers and device configurations, and the program creates network layouts based on these requirements. This system simplifies and speeds up the usually complex process of designing net- works, creating efficient structures that balance user needs with technical limitations. NetOptiAI helps organizations in various applications quickly build customized network infrastructures.

II. Problem Definition

A. Problem statement

Manual network design processes are often inefficient and struggle to meet diverse user priorities, making it difficult to scale and adapt to changing needs. NetOptiAI addresses this issue by streamlining the process, guiding users to create customized network topologies that align with their specific requirements.

B. Goals

- Automated Network Design: Create a system capable of interpreting user-defined network priorities to autonomously generate optimized network topologies. This will involve using advanced algorithms to automate the traditionally complex and time-consuming process of network design.
- User-Centric Interface: Develop an intuitive and user-friendly interface that allows users to easily articulate their network priorities. The interface will guide users through the process of defining their specific needs, such as server counts and device configurations, ensuring clear communication with the system.
- Efficiency and Scalability: Streamline the network design process, reducing time and resources required, while ensuring scalability to meet the demands of various network sizes and complexities.
- Resource Savings: Minimize the time and effort traditionally needed for network design, allowing organizations to allocate their resources more effectively.

III. Performance

Optimizing performance is crucial for the success of the NetOptiAI system, as users rely on it to efficiently generate network topologies and provide timely recommendations. Performance bottlenecks, such as slow response times, resource-intensive computations, and inefficient algorithms, can hinder user productivity and diminish the system's effectiveness. Continuous monitoring, performance testing, and optimization efforts are necessary to identify and address performance issues and ensure smooth system operation under varying workload conditions.

IV. Methodology

A. Approach

To achieve this, several natural language processing (NLP) techniques are employed:

- Tokenization: This process involves breaking down the user's text into individ- ual tokens, such as words or phrases. Tokenization helps in analyzing the text by segmenting it into manageable pieces.
- Regular Expressions (Re validator): Regular expressions can be used to iden- tify and validate specific patterns within the text, such as dates, numbers, or particular word sequences. This helps in cleaning the text and focusing on the most relevant parts.
- POS Tagging (Part-of-Speech Tagging): POS tagging involves identifying and labeling each word in the text

The_DT first_JJ time_NN he_PRP was_VBD shot_VBN in_IN the_DT hand_NN as_IN he_PRP chased_VBD the_DT robbers_NNS outside_RB ._.

first	time	shot	in	hand	as	chased	outside
JJ	NN	NN	IN	NN	IN	JJ	IN
RB	VB	VBD	RB	VB	RB	VBD	JJ
		VBN	RP			VBN	NN

Fig. 1. POS tagging

with its corresponding part of speech (e.g., noun, verb, adjective). This helps in understanding the grammatical structure of the text, which is essential for simplifying complex sentences.

- Rule-Based Parsing: This method involves defining a set of rules that describe how to extract relevant information about the network topology and system configuration from the user's input. These rules can be based on predefined patterns and heuristics, allowing the system to parse the input text and extract the necessary details (e.g., identifying the topology type and the number of systems).
- Data Extraction: Gather network details from user input, including the number of systems, topology type (e.g., star, mesh), and any nested configurations.
- Data Organization: Structure the extracted data hierarchically to reflect component relationships, such as a star topology with nested meshes showing central hubs and sub-topologies.
- JSON Formatting: Convert the organized data into JSON, a lightweight and human-readable format commonly used in web applications and APIs.

By combining these techniques, we can systematically analyze the user's prompt, simplify complex sentences, and produce a clearer and more accessible version of the text.

B. Re-gex

5.2.2 Reg-ex

Figure 5.1: Enter Caption

• Empty string (): Matches the empty string.

• Literal characters: Matches the character itself. For example, ”a” matches ”a”.

• Concatenation: If r1 and r2 are regex, then r1r2 is a regex that matches any string formed by concatenating a string that matches r1 and a string that matches r2.

(\ d+) system (w+ topology)

- (\d+): This part captures a sequence of one or more digits.
- system: This part matches the exact string ”system”.
- (\w+ topology): This part captures a sequence of one or more word characters followed by a space and the word ”topology”.

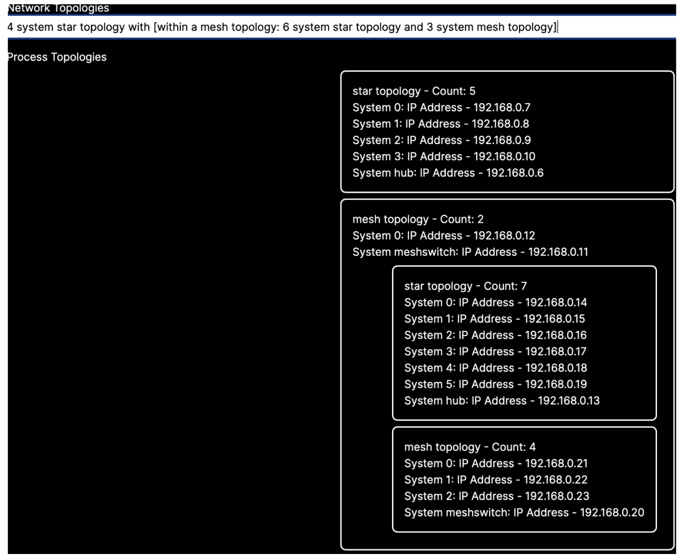

Fig. 2. Output

C. POS tagging

V. Implementation

1. User Interaction: Users engage with an intuitive frontend interface to specify their network requirements. The interface is designed to guide users through the input process seamlessly, providing feedback to ensure accurate data entry. This enhances user experience and improves input precision.

2. Frontend-to-Backend Communication: Once users submit their data, the frontend transmits this information to the backend API via HTTP requests. This raw data, detailing the network requirements, forms the basis for further processing and network design generation by the backend system.

3. NLP Preprocessing and Parsing: Upon receiving the user inputs, the backend API utilizes a Natural Language Processing (NLP) module to preprocess and parse the data. The NLP module extracts critical information, such as the number of systems, the type of network topology (e.g., star, mesh), and any nested configurations. This step ensures that the data is accurately interpreted and ready for network design.

4. Network Design Processing: With the parsed data, the backend network design engine creates optimal network configurations. The design engine evaluates various factors and constraints to generate solutions that are efficient and scalable. This ensures that the network design can adapt to changing requirements and future growth.

5. Backend-to-Frontend Communication: After completing the network design, the backend API sends the results back to the frontend interface. The frontend then displays these designs to the users, allowing them to review and, if necessary, refine the configurations based on their preferences or additional needs.

A. Result

The CLI details reveal the network's architecture and topology. The list of IP addresses shows the devices connected to the network, aiding in their identification and management. System labels like "System 0" and "System hub" clarify device roles within the network. The "7 system star topology" description indicates a central hub connecting multiple devices, with sub-topologies such as mesh and bus adding complexity. Topology counts, such as "Star topology - Count: 8" and "Mesh topology - Count: 5," provide insights into the distribution of topology types, helping administrators analyze and optimize the network.

ACKNOWLEDGMENT

NetOptiAI revolutionizes network infrastructure design by harnessing the power of artificial intelligence. This innovative system empowers users to define their network priorities, from server counts to performance metrics, enabling the AI to craft optimal network topologies. NetOptiAI delivers personalized solutions that balance user preferences and technical constraints, ensuring efficient, robust, and tailored network configurations.

REFERENCES

Please number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first . . ."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.