IntentLLM: An AI Chatbot to Create, Find, and Explain Slice Intents in TeraFlowSDN

IntentLLM: An AI Chatbot to Create, Find, and Explain Slice Intents in TeraFlowSDN

Daniel Adanza*, Carlos Natalino†, Lluis Gifre*, Raul Muñoz*, Pol Alemany*, Paolo Monti† and Ricard Vilalta*
*Centre Tecnologic de Telecomunicacions de Catalunya (CTTC), Casteldefells, Spain 08860
†Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden
Email: daniel.adanza@cttc.es

***Abstract*—A large language model (LLM) chatbot is integrated within TeraFlowSDN for intent manipulation. The resulting chatbot is capable of understanding the context and is able to carry out three actions: create, find, and explain intents using natural language while being flexible regarding the language used.**

***Index Terms*—large language models, software-defined networking, intent-based networking, network slices**

I. Key networking concepts

Large language models (LLMs) have revolutionized the way that natural language is processed and generated by artificial intelligence (AI) systems. In only a few years, many industries have successfully adopted such technology in many processes: machine translation [1], sentiment analysis [2], and text generation [3], making communication across languages and cultures more accessible and efficient than ever.

Despite the strong capabilities and the great popularity of the LLM, very few studies implemented this technology in the field of optical networks. Within them, it is worth highlighting [4], where the combination of text mining and a LLM results in an expert system able to understand the topic of materials used in optical networks and to answer properly using natural language. It is not surprising that its results seem to outperform the previous versions of the application.

Moreover, [5] also implemented a LLM as an expert system to provide technical information in a questions and answers system about alarm compression and fault analysis.

Alternatively, intent-based networking aims to simplify human interaction with the network control plane [**?**], where the human only defines what it needs and leads the way to achieve it (i.e., how) the incoming requests to the system. This means that the requester does not need to have any technical knowledge about the resources. For instance, in [6], the demonstration allowed users to perform a single operation (i.e., generate an intent) based on a set of requirements. This represents a promising use case, despite the fact that it does not integrate a LLM in itself, it takes advantage of a more simplistic natural language processing (NLP) to trigger a single operation which is the intent generation. However, it is important to remark that, to the best of our knowledge, no work takes advantage of LLMs for the manipulation of intents including creation, finding, and explaining intents.

This demonstration develops a new AI chatbot named *IntentLLM*, aiming at showcasing the potential of LLMs for the manipulation of intents. The LLM is integrated into a chatbot through which the user can create, find, and get explanations about existing transport network slices. The chatbot is also integrated with TeraFlowSDN, where slice creation prompts are executed, resulting in new transport network slices being deployed over the network. This demonstration will allow the audience to interact with the chatbot through natural language, and visualize the created transport network slices in the network.

II. Highlights of the innovation

This demonstration is the first to showcase how LLMs can be used not only to extract information from the network and act as an expert system, such as in [4], [5] but also to serve as an interface for network manipulation.

This is done by IntentLLM's interface with ETSI TeraFlowSDN which permits to create and perform new transport slice requests. User prompts in natural language are interpreted using the LLM and converted into TeraFlowSDN's transport slice requests.

The presented demo can be seen as an evolution of the previously presented work [**?**]. The inclusion of a LLM allows IntentLLM to be more flexible regarding the language used, e.g., by understanding context and synonyms, integrated into a more complete system. Additionally, new functionalities are added: The LLM has access to the list of existing intents and transport network slices objects and is able to carry on a search, find, and explain these existing elements or a part of them.

Finally, the audience will be able to inspect the creation of the resulting transport network slices over the network using the user interface offered by the TeraFlowSDN.

A. Innovation architecture

IntentLLM works as a unique component whose main goal consists of triggering different network requests to fulfil the needs from the customer. This component is built on top of a packet-optical network using a cloud-native SDN controller and is powered by the technologies illustrated in Figure 1 and its structure follows the model-view-controller architecture.

The user interacts with the chatbot built using Amazon Lex, which acts as the *view* in the architecture. The Amazon Lex component can be seen as an additional layer that communicates directly with the web UI component inside the TeraFlowSDN architecture. Internally, IntentLLM also uses

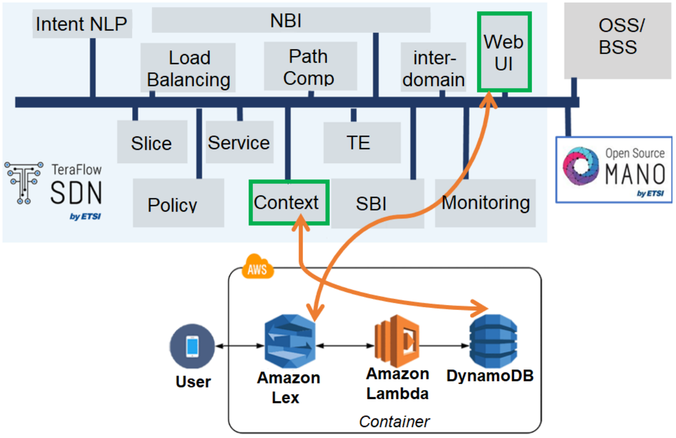

Fig. 1. Architecture of the demonstrator illustrating the different components of the solution and its integration with TeraFlowSDN.

a NoSQL database (i.e., DynamoDB) to store its context and to keep track of the current network state parsed from TeraFlowSDN's context, representing the *model* in the architecture.

The *controller* is implemented by an Amazon lambda function and its role consists of handling different tasks such as data validation, intent explanation, intent generation, intent provisioning, and communicating directly with the database. On the one hand, for user input interpretation, intent generation, and intent explanation, the lambda function communicates with the LLM hosted by Amazon. On the other hand, for intent provisioning, the lambda function communicates directly with TeraFlowSDN using the internal TeraFlowSDN gRPC messages to create new transport network slices in the network. Moreover, the lambda function also uses internal TeraFlowSDN messages to obtain and update the local database state for the existing intents/transport network slices.

Finally, it is important to mention that the solution presented can be easily adapted to communicate with other SDN controllers.

III. Description

The suggested solution encompasses a set of four steps executed sequentially:

Task identification: The algorithm has been fed with a subset of sixty sample sentences used as a training set to identify three types of operations based on the context: a) to explain an intent request using natural language, b) to create an intent by describing some requisites, and c) to retrieve information of the registered intents.

Request and validation of the key performance indicators (KPIs): As a second step, the chatbot asks for the required information which varies depending on the identified operation. To explain an intent request, the application requests the intent identifier and the part of the intent that would be explained leaving three choices to the user: To explain the locations, the network limitations, or both of them. Alternatively, to create an intent request, the user has to describe the requirements in terms of bandwidth limitation and locations simply by using natural language.

Finally, to consult the registered intents, the user will simply be asked if he wants to implement a filter and if appropriate will be asked to indicate the filtering criteria and the corresponding value.

Confirmation: As an intermediate step, the included information is been gathered and shown to the user for his approval.

Task execution: Finally, the required operation gets triggered and its functionality will vary depending on the requested operation.

A. *Triggered operations*

The chatbot allows three operations that can be executed depending on the task identification step described before:

To explain an intent: As a first step, the algorithm processes the information stored and extracts its main information. Regarding the location, the method extracts its unique universal global identifier (UUID), the switch and the port where it is placed for all the location points on the intent.

Alternatively, it extracts information regarding its constraint values in terms of required availability, requested capacity as much as bandwidth and time restrictions. As a main result, the most important information hierarchically stored in the intent is been extracted and processed.

As a second step, this extracted information is used to compose a message using NLP. For that purpose, a subset of templates is being used with a random selection to provide a more organic language.

To create an intent request: A detailed description of the process of intent generation based on a description provided by the user has been explained in detail in a previous demonstration [6], where a two-step process was presented: Firstly, the algorithm analyzes the user's description of requirements and implements an NLP algorithm to target relevant keywords regarding the location and bandwidth limitations. Secondly, the extracted information is used to construct hierarchically the intent request.

To consult the historically added intents:

To consult the registered intents, IntentLLM contains a representation of the current state of the transport network slices. This allows IntentLLM to apply filters if necessary. For filtering, the user needs to include the attribute name and the value of the filter. Some examples could be: Filtering intents on *"port: 2000"* or *"with a duration of 2 hours"*.

IV. Items to be demonstrated

A. *physical space and logistics*

This demonstrator will allow attendees to interact with the AI-based chatbot in real-time. Hence, to present the demo, physical space to place a computer will be needed to perform the demonstration. The logistics for this project will be pretty simplistic since only a laptop would be required to perform the demonstration.

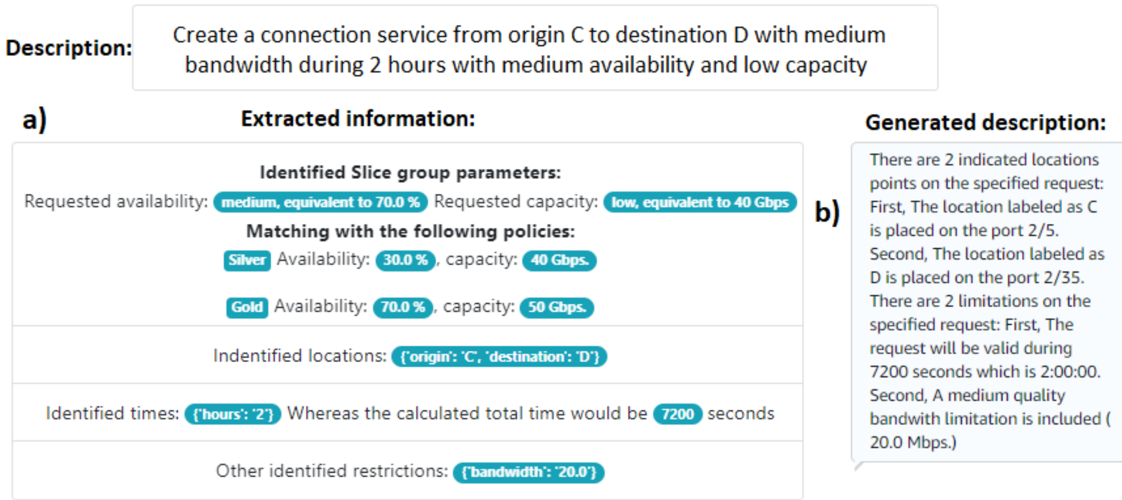

Fig. 2. Figure illustrating some sample results shown to the user.

B. How the attendees interact with the demonstration

The attendees will be able to describe an intent using natural language. They will just simply need to order it and the chatbot understand what they want depending on the context. Secondly, the application will ask for the required information and wait for the user's confirmation. Finally, the request is submitted to TeraFlowSDN, and its realization can be visualized through a Web interface.

Alternatively, the user can also request an intent to be explained just by indicating the intent identifier and the part of the intent that would like to be explained. As a main result, the attendees will be able to see different explanations using clear, understandable and organic language. IntentLLM then consults the deployed intents and implements filters into the data just by using natural language.

C. Sample results

Figure 2 shows two examples of how the attendees will be able to interact with IntentLLM.

The top of Figure 2 shows a description of a particular case where a set of requirements are specified using natural language such as *"from origin C to destination D"* or *"during 2 hours"*.

On the one hand, Fig. 2(a) illustrates some results after carrying on the generate intent operation based on the given description. The system was able to extract and parse information such as time restrictions which would be 7200 seconds and to match the requirements with different available service layer agreement (SLA) policies.

On the other hand, Fig. 2(b) shows the generated description of the same intent, after executing the intent explainer method, where it is possible to see that the chatbot is able to explain the locations and the bandwidth limitations based on the given intent request parameters.

V. Conclusion

The presented demonstration brings proof that the LLM are a very powerful technology that, when combined with intent manipulation is capable of simplifying and automating many ordinary tasks that are needed to maintain and configure the network infrastructure. This integration suggests an enhanced user experience within TeraFlowSDN, as users can interact with the system using natural language commands for intent-related tasks.

Acknowledgment

This work was partially supported by the Horizon Europe Hexa-X-II (101095759); Ministerio de Asuntos Económicos y Transformación Digital and the European Union-Next GenerationEU in the frameworks of the Plan de Recuperación, Transformación y Resiliencia and of the Mecanismo de Recuperación y Resiliencia through UNICO-5G I+D 6GMICROSDN project under references TSI-063000-2021-19, TSI-063000-2021-20, TSI-063000-2021-21; MCIN/AEI/10.13039/501100011033/FEDER/UE and by ERDF A way of making Europe through the Spanish and by Sweden's innovation agency VINNOVA within the EUREKA cluster CELTIC-NEXT project AI-NET-PROTECT (2020-03506).

References

[1] H. Xu *et al.*, "Monolingual denoising with large language models for low-resource machine translation," in *Natural Language Processing and Chinese Computing,* F. Liu *et al.*, eds. (Springer, 2023), pp. 413–425.

[2] N. E. Marref, "Talking taboo topics with ChatGPT AI: Bais/sentiment analysis study/personality analysis," Ph.D. thesis (2023).

[3] J. Li *et al.*, "Pretrained language models for text generation: A survey," arXiv preprint arXiv:2201.05273 (2022).

[4] J. Zhao *et al.*, "OpticalBERT and OpticalTable-SQA: Text- and table-based language models for the optical-materials domain," Journal of Chemical Information and Modeling **63**, 1961–1981 (2023). PMID: 36940385.

[5] Y. Wang *et al.*, "AlarmGPT: An intelligent operation assistant for optical network alarm analysis using ChatGPT," in *European Conference on Optical Communication (ECOC),* (2023), p. Th.A.1.2.

[6] R. Vilalta *et al.*, "Demonstration of intent-based networking for end-to-end packet optical cloud-native SDN orchestration," in *European Conference on Optical Communication (ECOC),* (2023), p. D3.