Recent Advances in Intent-Based Networking: A Survey

Engin Zeydan* and Yekta Turk$^{\diamond}$
*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Castelldefels, Barcelona, Spain, 08860.
$^{\diamond}$Mobile Network Architect, Istanbul, Turkey, 34396.
engin.zeydan@cttc.cat, yekta.turk@ieee.org

***Abstract*—This paper investigates the recent-advances in intent-based technologies while concentrating on aspects related to network management and orchestration. We provide a comprehensive analysis of the standardization activities as well as platforms related to intent-based networking. At the end of the paper, we also provide some insights into challenges related to future development process on the intent-based networking design. Our survey results indicate that intent-based networking concept has not evolved further since 2015 in terms of framework, platform and tool developments. However, recent rapid advances in Natural Language Understanding (NLU) propelled by IT and cloud giants (Google, Amazon, Facebook) are expected to increase its adaption into networking and telecommunication world in the forthcoming years.**

Keywords*—*intent, network, management, orchestration, survey.

I. Introduction

Currently, most the operators are involved in complex network provisioning steps where the configuration and implementation updates are closely related to underlying heterogeneous and diverse infrastructure. This brings a necessity to build abstraction layer on top the network infrastructure where operators can tune parameters agnostic of the diverse infrastructure. The main motivation of building an higher-level management abstraction over complex, heterogeneous and distributed networks has paved the way for experimenting solutions of intent-based networking infrastructures over the recent years. At the same time, as interest in developing applications using Artificial Intelligence (AI)/Machine Learning (ML) platforms has risen over last decade due to recent advancements on deep learning mainly with applications in Natural Language Processing (NLP) [1] domain, the surge for intent-based networking is also expected to resurrect after a dormant period.

Advances in Natural Language Understanding (NLU) systems together with neural network based algorithms such as Bidirectional Encoder Representations from Transformers (BERT), Robustly Optimized BERT pretraining Approach (RoBERTa), General Language Understanding Evaluation (GLUE) and Enhanced Representation through kNowledge IntEgration (ERNIE) have advanced the knowledge to convert the user queries expressed in a given language (e.g. in English) into a representation that is adequately structured so that it can be processed by an automated service [1]. By abstraction of network services, ever growing complexity of the network service management as well as corresponding cost levels are envisioned to be curbed via intents. Recent advancements in AI can be an important enabler to develop network management solutions that are driven by intents. Considering the advanced integration of intent-based technology into existing management systems, a three-layered architecture can be designed as depicted in Fig.1. The main motivation of this architecture is to introduce intent-based automation to gain new abilities that would not be in reach with human workforce. In the business layer, the intents are based on Key Performance Indicators (KPIs) and includes specific Service Level Agreement s (SLAs), processes, goals, targets or objectives that are triggered by users from the business layer. The objective of intent layer is not to execute the full planned sequence of actions blindly, but re-evaluate and re-plan after every step. The proactively planned actions must be subjected to approval inside this layer.

In the intent layer of Fig.1, *Knowledge* handles abstraction of intents. It also provides the reasoning for intents by building relationship between objects. *Agent* in the intent layer provides an interface to the network objects and performs actions on the network objects after evaluating the intents. The *Agent* does not have intelligence and it sends requests to *Knowledge* for reasoning when action is required. The *Knowledge* informs the *Agent* with the composition which is the result obtained from the analysis of intents. *Data* observes the network objects and used for effective storage. *Data* has the network topology and inventory information. *Data* is mainly responsible for forwarding updates to the *Knowledge* in a dynamic manner whenever a new objects is deployed in the network, an object is de-installed or a topology change occurs. *Data* provides the modelling of the network topology and transfer it to the *Agent*. The most important state of *Data* realizing the moving target and updating the model with more accurate evaluations or if the business intent is modified. At the bottom of Fig. 1, there is the network layer that is containing the physical nodes, namely objects. Its responsibility is to transform the network data into an formal representation so that intent layer can easily work with. The abstract model of the hardware is stores in this layer and it is responsible to execute actions requested by the *Agent*.

The term "intent" have been in telecommunication industry over the last 15 years [2]. It is basically adopted as an evolved version of the term "policy" which dates back to Sloman's policy-driven management system in 1994 [3]. The motivation for this evolution was that policy management was hard and different entities (e.g. end-users with no technical insights, app developers that are developing network services without complicated network interface experience and know-how, operators that are willing to initiate network services in more abstract and robust manner) wanted much simpler solutions.

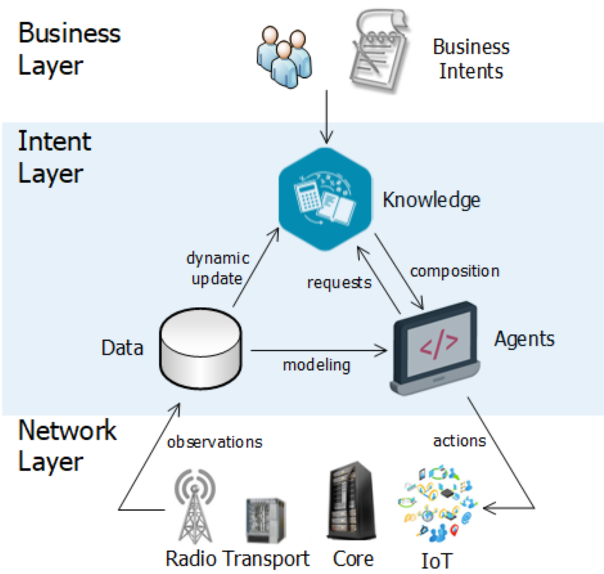

Fig. 1: Illustrative intent-based networking architecture.

Intents are defined as a set of specific policy types that are written in high level operational and business objectives [4]. They are mainly written by humans and should be transformed into device consumable or lower-level forms. The main idea behind creating intent-based technologies is to meet the system requirements without detailing how to achieve these objectives. Moreover, they are designed to be small in size. Policies are mainly used to refer to events, conditions and actions which can include a wide scope. On the other hand different than policies, intents are mainly used to allow for rules to choose the behaviour of a system and are considered to be a subset or type of policy.

Internet Engineering Task Force (IETF) draft in [5] is one of the few studies that have concentrated on definitions of policy and intent and their corresponding differences. Policy is commonly associated to Event-Condition-Action (ECA). Intents however has been defined as abstract and high-level policy that are used for network operation in [6]. Hence, intents can be thought as high-level business or operational targets that a system should meet. However, the specifications to achieve these targets are not given, i.e. without any given characteristics of ECA. For example, consider the case of a route establishment. A **policy** can be *when a route establishment request between nodes A and B comes in to network administrator, if the user is C, block the traffic*. Then, an **intent** can be defined similarly as *Routing between nodes A and B for user C is prohibited*. Therefore, the way to enforce the intent is left to the system.

Some important aspects to consider as characteristics of intent are [2], [7]: (i) intents are agnostic to underlying hardware and should be portable across technologies. (ii) intents provide context and are suitable to build a non-conflicting service deployment. (iii) the intents can be persistent (connect into database) or transient (transform data into another format). (iv) the intents are compatible and the requirements are more explicit and simplified using intents. In network management and orchestration domain, the first real necessity to design the infrastructure towards intent-based architecture came along with the introduction of Software Defined Networking (SDN) and OpenFlow protocol back in 2014. This was mainly driven by the requirements of higher-level Application Programming Interfaces (APIs) where open-source SDN-based projects such as Open Network Operating System (ONOS) [8] and OpenDayLight [9] started to work on intent-based APIs.

Our main observations during surveying recent works on intent-based networking can be summarized as follows. Currently, the recent frameworks and platforms that provide intent-based networking are mainly focused inside academia and not by industry. Therefore, commercial development and deployment of intent driven network management services have not emerged as yet. On the other hand, major advancements in AI techniques especially in NLP/NLU areas are expected to influence the transfer of this knowledge into commercial products and services in telecommunication and networking domain in the coming years. Moreover, although various standardization efforts have emerged during the last decade together with the introduction of SDN and OpenFlow, the intent driven network management concept has not progressed as expected. Therefore, the standardization efforts are currently at their infancy period. Additionally, the representation or inference of knowledge from intents are still a major challenge in both standardization and platform development processes.

The rest of the paper is organized as follows: In Section II we discuss about the recent standardization activities. In Section III, we discuss about the developments on intent-based networking platforms. In Section IV, we discuss about the outcomes of the survey and give future directions. Finally, in Section V, we give conclusions.

II. Intent-Based Networking Efforts in Standardization

The 3rd Generation Partnership Project (3GPP), European Telecommunications Standards Institute (ETSI), Open Networking Foundation (ONF) and International Telecommunication Union (ITU) have all developed their own study groups on intent-based networking. The first standardization effort for intent-based networking was in ONF back in 2016 [10]. The main idea was to create an intent NorthBound Interface (NBI) handler that is embedded into or external to network controller. The Boulder project imitated by ONF to provide Open Intent NBI. ONF efforts have now merged with ON.Lab (founders of ONOS controller) which have their own intent-based interface. The work on TR:28.812-"Study on scenarios for Intent driven management services for mobile networks" in Release 16 in the scope of SA5 is 3GPP's effort on intent-based network management that started back in 2018 and is still an ongoing effort [11]. This document asserts the concept of Intent Driven Management (IDM) where an Intent Driven Management Service (IDMS) is provided to consumers to manage 5G network and services. Utilization of intent driven management service is envisioned to originate from communication service providers/customers and network operators in the considered scenarios. Some of the considered scenarios are related to intent driven service deployment, network provisioning, network optimization, coverage and capacity management.

ETSI has initiated the Zero-touch network and Service Management (ZSM) working group [2] for describing means network automation in 2018. The document provides details on automation in network management and also concentrates on policy-driven automation, intent-based automation as well as intent-based service orchestration in three separate chapters. This standard document also argues that intent can be applied to automation in different layers, in APIs, network systems or service orchestration levels. Another working group within ETSI under ETSI Experimental Networked Intelligence (ETSI ENI) considers intent as part of a larger policy classification (where the other policy classifications are declarative and imperative policies) [12]. In this standardization report, imperative policies are simply condition-action (CA) or ECA tuples. Therefore, the order of execution of statements is important. Declarative policies on the other hand, try to achieve a specific goal by expressing what need to be done instead of defining its implementation details. The order of execution of statements is irrelevant. Intent policies are in fact similar to declarative policies in the sense of concentrating on what to do instead of how to do. On the other hand, they are expressed in much small and simpler policies (3-30 lines) working together and requires implementation of a translation process, i.e. mapping (possibly using NLP techniques in practical cases).

ITU-T Study Group 13 explores intent as a declarative mechanism (written in ML meta-language) where technology-agnostic ML use case can be deployed by operators inside their focus group on Machine Learning for Future Network including 5G (FG-ML5G) [13]. In that sense, intents are used as high level ML pipeline components. However, building this meta-language is also foreseen as one of the main challenges in future implementation of intent-based networking. In summary, all ongoing standardization activities take intents are high level specified goals without any specifications on how to execute the concrete actions. Hence, policies can be derived from these intents. In some cases such as in ETSI ZSM and ETSI ENI, intents can also be expressed by human language and later to be translated into models that can be interpreted by machines. One of the major challenges that is expressed in standardization documents is the representation of intent in terms of language and model specifications which is still an ongoing and important effort.

III. Developments on intent-based platforms

Most of the related solutions on frameworks and platforms on intent focus on conversion of high-level descriptions into lower-level configurations, i.e. translatable intents. An intent is envisioned to translate *what* into *how* after its activation and is continuously monitored to enforce this cycle [4]. Similarly, another approach to intents is to express them as guidelines to shape the actions (transform, filter, adapt, etc) of the devices on a system. This is commonly referred as consultative intents. The main difference between translatable and consultative intents arises when feedback mechanism is adapted inside the system with the consultative intents. Considering the historical and current status of the system, the enforced and monitored intents are also adapted. This is commonly applicable for distributed system management where distributed agents need to adapt to the dynamism of the system. Consultative and translatable intents are also referred to as high granularity and low granularity respectively in some related works [7]. Some other prior works have also used these two types of intents together to build an intent-based SDN system [14].

Low granular intents: These intents are mainly used only for translatable purposes. From a practical direction frameworks such as ONOS and OpenDayLight SDN controllers have been working on developing APIs of intent-based networking. ONOS defines the intent as an immutable model object that is requested by application to its core and is used to alter the behaviour of the network in terms of how it operates [8]. ONOS also defines the terms domain intents, which can be applied to third party controllers that do not directly observe the devices that are external to their domains. ONOS intent implementation has also been used or inspired for intent-based developments in other related works such as using NLP to specify connectivity requests and convert into ONOS intents in [15], reconciliation of several intents on top of infrastructure to enable traffic engineering and optimization of intent framework in [16], enabling a separate intent framework that can bridge intent expressions across several domains via including both KPI constraints and high-level connectivity intentions in [17] or high level intents translated into ONOS controller that consider encryption, bandwidth as well as domain constraints in [18].

OpenDayLight framework also includes sub-projects (e.g. Network Intent Composition (NIC) [19], NEtwork MOdeling (NEMO) language in [20] and Group-based Policy (GBP) abstractions in [21]) that are involved with intent-based networking back in 2014 and 2015. For example, NEMO uses a translation engine to process network languages so that modules such as OpenFlow can be rendered underneath. Similarly, intent based NBI of NIC allow descriptive way to obtain what is desired from the network using any kind of protocols, including OpenFlow, BGP, Netconf, etc. GBP framework is designed to provide APIs to capture user intents. However, those projects are mainly inactive in recent years. Some related IETF drafts [22], [23] and language extension papers [24] on NEMO has also been published but the further enhancements are not further investigated. There exists other controllers that are been used in the context of intent-based networking such as a framework called iNDIRA (Intelligent Network Deployment Intent Renderer Application) in [25], Propane framework in [26] applied to data center and backbone networks, MD-IDN for multi-domain environments in [27].

High granular intents: The main idea of high granular (or consultative) intents is not only translate intents but also receive feedback that affects these translations and adapt accordingly. This approach is applied to industrial networks in [28]. Another form of application of this concept can also be found in social network graphs. Finding the right intent of a user query on a social network graph is studied in [29]. This approach uses network representation learning which uses ML algorithms to represent networks appropriately which can be useful to solve implicit context in intent-based networking. An intent driven networking (IDN) concept is coined in [30] where hierarchical strictures of mediators are used to keep track of the intent status and realign accordingly. Software developed by VeriFlow [31] aims to develop the intent-based networking from a different perspective by focusing on precisely formalizing network behavior by means of measurements and then verifying it against user intent to avoid any deviations.

TABLE I
CHARACTERISTICS OF LAYERS IN INTENT-BASED NETWORKING ARCHITECTURE AND CORRESPONDING RELATED WORKS.

Layer		Characteristics	Related Works
Business Layer		— Higher-level declarative policy that operates at the level of a network and services	[3], [22], [24]
		— Provide semantic to consume network resources	[6], [31]
		— Allowing high-level guidance by a central entity	[4], [5], [23]
		— Detect and resolve conflicts between multiple intents	[7], [8]
Intent Layer	Knowledge	— Access to knowledge and execute judgement	[1], [2]
		— Performs inference from relations between objects	[18], [27]
	Agent	— Capture the business intent and translate into policies	[17], [21], [23]
		— Utilize ontology-based approach to communicate with users	[25]
		— Communication interface directly to the network objects	[9], [10], [11]
	Data	— Keep the state of each intent and the relation between network objects	[16], [26], [29]
		— Provides models for the observed data	[13], [20]
		— Provides algorithms for data modeling	[30]
Network Layer		— Present the abstraction of domain-specific data and control plane technologies	[12], [19], [32]
		— Specify context-aware architecture for enhancing the network intelligence	[14], [15], [28]

NLU developments for intents: Recent years have witnessed tremendous efforts and advances in terms of learning representation of language across a range of diverse domains and tasks. To evaluate various emerging models, GLUE benchmark has been designed [1]. These benchmark tests are aimed to get closer to human level understanding of the algorithms and techniques. For this purpose, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) (LSTM, GRU, etc.)-based architectures and their variations (RNN seq2seq, CNN seq2seq, seq2seq models with Attention, BERT, etc.) are trained as a baseline to perform various text analysis operations on dataset for intent recognition, classification or slot tagging tasks. For instance, the authors in [32] have used BERT for training intent classification and slot filling tasks jointly. On the other hand, in industry Apple's Siri or Amazon's Alexa are some examples of the mostly utilized related products that are using those advanced NLU models and techniques.

IV. DISCUSSIONS, CHALLENGES AND FUTURE DIRECTIONS

From the recent advances in intent-based networking, we can extract the following observations: First, differentiating between event-condition-action policies and the intents is simplified by concentrating on "what" rather than "how". Second, intents and their specification languages are also defined separately and corresponding frameworks are developed within their own use cases. This shows that further efforts on framework developments are necessary to merge into similar intent specification languages. Third, as the abstraction level is increased, e.g. at business layer of Fig. 1, more consultative intents will be generated. As the intents are progressed down towards network layer, stronger translatable intents will be required. Therefore, the co-existences of these two types of intents can be necessary in some use cases.

There are also multiple challenges during specifications of intents. The first is to be able to represent intents so that it can processed and enforced. This requires development of abstraction of intents as general graphs or trees. Ontologies and graph databases can be used together to build intent semantics. In case there exists many intents where some can be conflicting with each other, elimination of those conflicts and ensuring increased negotiations between intents between business, network and intent layers is another related challenge. To convert intent, translation of human specific language into machine language is necessary. Recent advances in abilities of general language understanding using deep neural networks have come a long way. NLP is a rapidly growing research area and new algorithms and techniques emerge constantly. Intent classification (that focuses on predicting the intent of the user query) and slot filling (used to extract semantic concepts) tasks in NLU are some relevant examples in this domain [32]. This has also increased the chance of adoption of these technologies and ideas into intent-based networking domain.

Although recent developments in AI technologies (especially in NLP) can help in bringing this gap closer, the hidden ambiguities and context in human language is still an impediment for this transformation. Implicit context which can be clearly understood by humans but not by machines pose challenges to intent-based networking implementation. Lack of human labeled data in NLU and NLP tasks is also a major impediment towards generalization capability of these techniques. Moreover, most of the pre-built models used in NLU tasks are built in English language and can also be biased depending on the collected data used for training the model. Collecting and annotating a similar corpus as big as English in another language requires thousands of samples which need to be both representative and diverse. Depending on the problems, computational resources can also be significant. Some models based on NLU tasks require excessive memory and computationally intensive which can be cumbersome to deploy in small devices, e.g. on mobile devices. Depending on the use-case, instead of relying on more complex design simpler architectures can be utilized during NLU process. This can provide better trade-off between speed and accuracy. Finally, privacy of user intents is another issue that need to be addressed within the framework development processes.

As a summary of related works corresponding to each layer described in intent-based architecture of Fig. 1, we described the characteristics of the layers as well as the studies that are related with each these characteristic in Table 1. As a future direction, expressing intents concretely in the definitions of languages and building models that can manipulate and enforce intents effectively over various domains are some important study items that need to be addressed in both standardization activities and framework/platform development processes. Moreover, continuous/active learning paradigm can also be introduced as a service to learning agents through a state action model.

V. Conclusions

In this survey paper, we provided recent advances in intent-based networking concentrating on network management and orchestration. We first provide a comprehensive analysis of existing frameworks and platforms and later concentrate on providing the latest activities in standardization domain. Finally, we discuss about the challenges and future directions towards building an intent-based networking system for the telecommunication networks. Our review analysis indicate that even though the concept of intent-based networking has been around over the last decades, no major updates have been observed over the last years. On the other hand, together with the emergence of AI technologies applied in NLP and NLU domains, major improvements in intent-based advancements can be foreseen to be transferred into networking and telecommunication world in the upcoming years.

Acknowledgment

This work was funded by Spanish MINECO grant TEC2017-88373-R (5G-REFINE) and by Generalitat de Catalunya grant 2017 SGR 1195.

References

[1] GLUE, "GLUE Benchmark Leaderboard." https://gluebenchmark.com/leaderboard, 2020. [Online; accessed 17-Jan.-2020].

[2] ETSI, "Zero Touch Network and Service Management (ZSM) Means of Automation." https://bit.ly/2THSfRl, 2018. [Online; accessed 17-Jan.-2020].

[3] M. Sloman, "Policy driven management for distributed systems," *Journal of network and Systems Management*, vol. 2, no. 4, pp. 333–360, 1994.

[4] Cisco, "Intent-Based Networking: Building the bridge between business and IT." https://bit.ly/2In0D3i, 2018. [Online; accessed 17-Jan.-2020].

[5] A. Clemm, L. Ciavaglia, L. Granville, J. Tantsura, "Intent-Based Networking - Concepts and Overview." https://bit.ly/2ImukBF, 2020. [Online; accessed 17-Jan.-2020].

[6] M. Behringer, M. Pritikin, S. Bjarnason, A. Clemm, B. Carpenter, S. Jiang, L. Ciavaglia, "Autonomic Networking: Definitions and Design Goals." https://tools.ietf.org/html/rfc7575, 2015. [Online; accessed 17-Jan.-2020].

[7] C. Li, O. Havel, W. Liu, P. Martinez-Julia, J. Nobre, D. Lopez, "Intent Classification (IETF Network Working Group)." https://bit.ly/3azxuhu, 2020. [Online; accessed 17-Jan.-2020].

[8] ONOS, "Open Network Operating System: Intent Framework." https://bit.ly/2wvXdbZ, 2014. [Online; accessed 17-Jan.-2020].

[9] OpenDaylight, "OpenDaylight Project." https://www.opendaylight.org, 2013. [Online; accessed 17-Jan.-2020].

[10] ONF, "ONF TR-523- Intent NBI – Definition and Principles ." https://bit.ly/2TBYxSc, 2016. [Online; accessed 17-Jan.-2020].

[11] 3GPP Technical Specification Group Services and System Aspects, "Study on scenarios for Intent driven management services for mobile networks, Telecommunication management." https://bit.ly/2vJ2aOr, 2019. [Online; accessed 17-Jan.-2020].

[12] ETSI Experiential Networked Intelligence (ENI), "Context-Aware Policy Management Gap Analysis." https://bit.ly/2PTfwP1, 2018. [Online; accessed 17-Jan.-2020].

[13] ITU-T Focus group on Machine Learning for Future Networks including 5G (FG-ML5G), "Unified architecture for machine learning in 5G and future networks." https://bit.ly/3czPha0, 2019. [Online; accessed 17-Jan.-2020].

[14] H. Zhang, Y. Wang, X. Qi, W. Xu, T. Peng, and S. Liu, "Demo abstract: An intent solver for enabling intent-based sdn," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 968–969, IEEE, 2017.

[15] P. Sköldström, S. Junique, A. Ghafoor, A. Marsico, and D. Siracusa, "Dismi-an intent interface for application-centric transport network services," in *2017 19th International Conference on Transparent Optical Networks (ICTON)*, pp. 1–4, IEEE, 2017.

[16] D. Sanvito, D. Moro, M. Gullì, I. Filippini, A. Capone, and A. Campanella, "Onos intent monitor and reroute service: enabling plug&play routing logic," in *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, pp. 272–276, IEEE, 2018.

[17] W. Cerroni, C. Buratti, S. Cerboni, G. Davoli, C. Contoli, F. Foresta, F. Callegati, and R. Verdone, "Intent-based management and orchestration of heterogeneous openflow/iot sdn domains," in *2017 IEEE Conference on Network Softwarization (NetSoft)*, pp. 1–9, IEEE, 2017.

[18] T. Szyrkowiec, M. Santuari, M. Chamania, D. Siracusa, A. Autenrieth, V. Lopez, J. Cho, and W. Kellerer, "Automatic intent-based secure service creation through a multilayer sdn network orchestration," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, no. 4, pp. 289–297, 2018.

[19] OpenDaylight, "Network Intent Composition." https://wiki.opendaylight.org/view/Network_Intent_Composition:Main, 2015. [Online; accessed 17-Jan.-2020].

[20] OpenDaylight, "Network Modeling language (NeMo)." https://bit.ly/2TDBh6w, 2015. [Online; accessed 17-Jan.-2020].

[21] OpenStack, "Group-Based Policy for OpenStack." https://bit.ly/2PU3HrN, 2015. [Online; accessed 17-Jan.-2020].

[22] S. Hares, "Intent-Based Nemo Overview." https://bit.ly/2VWTkqS, 2015. [Online; accessed 17-Jan.-2020].

[23] S. Hares, "NEtwork MOdeling (NEMO) Language." https://bit.ly/3azdKul, 2016. [Online; accessed 17-Jan.-2020].

[24] Y. Tsuzaki and Y. Okabe, "Reactive configuration updating for intent-based networking," in *2017 International Conference on Information Networking (ICOIN)*, pp. 97–102, IEEE, 2017.

[25] M. Kiran, E. Pouyoul, A. Mercian, B. Tierney, C. Guok, and I. Monga, "Enabling intent to configure scientific networks for high performance demands," *Future Generation Computer Systems*, vol. 79, pp. 205–214, 2018.

[26] R. Beckett, R. Mahajan, T. Millstein, J. Padhye, and D. Walker, "Don't mind the gap: Bridging network-wide objectives and device-level configurations," in *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 328–341, ACM, 2016.

[27] S. Arezoumand, K. Dzeparoska, H. Bannazadeh, and A. Leon-Garcia, "Md-idn: Multi-domain intent-driven networking in software-defined infrastructures," in *2017 13th International Conference on Network and Service Management (CNSM)*, pp. 1–7, IEEE, 2017.

[28] B. K. Saha, D. Tandur, L. Haab, and L. Podleski, "Intent-based networks: An industrial perspective," in *Proceedings of the 1st International Workshop on Future Industrial Communication Networks*, pp. 35–40, ACM, 2018.

[29] Z. Meng and H. Shen, "Scalable aspects learning for intent-aware diversified search on social networks," *IEEE Access*, vol. 6, pp. 37124–37137, 2018.

[30] Y. Elkhatib, G. Coulson, and G. Tyson, "Charting an intent driven network," in *2017 13th International Conference on Network and Service Management (CNSM)*, pp. 1–5, IEEE, 2017.

[31] VeriFlow, "A Proactive Approach to Assuring Network Resilience and Protection." https://bit.ly/2PTZILR, 2017. [Online; accessed 17-Jan.-2020].

[32] Q. Chen, Z. Zhuo, and W. Wang, "Bert for joint intent classification and slot filling," *arXiv preprint arXiv:1902.10909*, 2019.