Distributed Processes for Spoken Questions and Commands Understanding

Dario Di Mauro
University of Naples "Federico II"

Antonio Origlia
University of Padua

Francesco Cutugno
University of Naples "Federico II"

{dario.dimauro,cutugno}@unina.it antonio.origlia@dei.unipd.it

Abstract

Commercial products labelled as smart devices usually recur to a centralised system that processes all the requests. A distributed model, where nodes independently interact with the environment, may provide a widespread support for both users and other devices. In the latter setup, each entity has a partial awareness about defines the requests accepted by the network, and this aspect complicates the task. This paper improves an existing distributed model, called PHASER, by proposing linguistic analysis techniques to manage non-matching requests. NLP methods produce a confidence; any PHASER node forwards non-matching requests to close peers. PHASER exploits the confidence to rank the adjacent peers and deliver the question to the best node. Partial Matching and a *Bag-of-Words* models will be compared with the currently adopted full matching. The Bag-of-Words approach offered the best results in terms of both quality and required time.

I prodotti commerciali, etichettati come dispositivi intelligenti, di solito usano un sistema centralizzato per processare le richieste. Un modello distribuito, dove i nodi interagiscono indipendentemente con l'ambiente, può fornire un supporto più ampio per gli utenti. Nel secondo setup, ogni entità è parzialmente a conoscenza delle richieste accettate da ogni nodo del network. Questo lavoro si propone di migliorare un modello distribuito esistente, chiamato PHASER, ricorrendo a tecniche di analisi linguistiche per gestire le richieste non accettate localmente; ogni nodo PHASER inoltra queste richieste ai nodi adiacenti. Metodi di NPL producono una confidence*; PHASER la sfrutta per ordinare i nodi vicini e inoltrare la richiesta al migliore. Modelli basati su* partial matching *e* bag-of-words *saranno confrontati con il sistema attualmente adottato, basato su* full matching*. Dal confronto,* bag-of-words *ha riportato i risultati migliori sia di qualità che di tempo necessario per la risposta.*

1 Introduction

This paper introduces a new distributed approach to question answering and command execution in different Intelligent Environments (henceforth IE). This idea is rarely encountered in literature (with a few exceptions by Surdeanu et al. (2002)). IE is a new discipline including Domotic, Internet of Things, Cultural Heritage Technological Innovation and other similar issues. In our approach, devices in the environments constitute nodes of a network and this network provides services to an interacting user. Reasons of interests for NLP studies in this kind of application lie in the idea that user requests are delivered using Natural Language (mainly speech) in the simplest case, or multimodally by integrating speech with gestures and interaction with physical controls.

Nowadays, smart devices commonly propose interaction through natural language to the users. As the services offer becomes wider, a network of specialised applications managing very specific domains is masked behind a single named character (Alexa, Siri, Cortana). While this is common for single devices hosting multiple applications, which inform the operating system about their capabilities through dedicated languages (e.g. SRGS[1], Hunt and McGlashan (2004)), the

[1] https://www.w3.org/TR/speech-grammar/ retrieved on October 2017

model is being transferred to networks of different devices. Optimising the communication between these devices is a critical issue to reduce response times, balance communication and improve quality of service. In this paper, we will concentrate on showing how the *confidence* metric, commonly available in NLP techniques, can be exploited to support rapid adaptation of the network to request dispatching, avoiding broadcasting.

We will mainly describe the simplest case of speech interaction and understanding; for a view on the complex multimodal approach, please refer to Valentino et al. (2017). Given these premises, nodes in the given network are able to respond to simple questions or commands uttered by the user. In a first approach, utterances should conserve a coherence with the "nature" of the node, i.e. if I am "talking" to the kitchen or to the microwave oven, I should make requests strictly inherent with the device functions. On the contrary, we wish to expand the "intelligence" of the environment giving to the user the possibility to make any kind of requirements to any node in the network. In this view, each node is able to classify the string deriving by the speech utterance assigning it to one of the many classes of relevant action the environment can realise, even if the node itself is not able to execute that action. The introduction of a distributed knowledge base and of network information spreading techniques concur to the realisation of an environment extremely reactive, scalable and easily configurable for different domains. The system is reactive as the network connections are strongly optimised: redundant and rarely used paths are pruned. Mechanisms for knowledge distribution are optimised in order to deliver the proper answer minimising network reaction times. The system is easily configurable to different domains as this kind of networks just require a formal description of the semantics of each node, of the action classes they are able to process, and of the most probable connection among nodes that *a-priori* the environment designer implements. In order to realise this system, many NLP software modules are needed, and among these: an automatic spoken dialogue manager, a Spoken Language Understanding system, an ontology modelling the environment and the devices. An extended description of each part can be found in Di Mauro et al. (2017); this paper focuses on a linguistic analysis to improve the navigation of the request through the network.

In Section 2 we present related works; Section 3 recalls the model of our system. In Section 4 we discuss a network of interactive entities, highlighting differences about the current version and the contribution of this paper. Experiments and Discussion are presented in Sections 5 and 6. Section 7 concludes the paper.

2 Related works

Our idea is to provide a distributed network of entities, where each node interacts with the user through multimodal interaction. Knowledge is local to the node and limited to the provided services. If the node is not able to produce the expected output for a request, it sends the message to others in the network, without a prior determined target node. The intelligence perceived by the users is built upon a collection of partial nodes' intelligence. This system, called PHASER, has been firstly introduced by Di Mauro et al. (2017).

Distributed approaches for Human-Computer Interaction have been widely discussed in literature. Multi-Agent Systems have been applied to smart environments by Li et al. (2016), Pajares Ferrando and Onaindia (2013) ; their work is based on the discovery of semantic resources and orchestration, with negotiation between user and devices. Valero et al. (2016) proposed a system with multiple users, various roles and access policies.

The goal of this work is to provide a strategy to better rank close nodes according to the exposed information about the accepted inputs. By considering the Navigation problem from a Question/Answering (Q/A) point of view, PHASER could be theoretically compared with distributed Q/A systems (Surdeanu et al., 2002). Q/A systems do not collect entire documents, but they extract just short and relevant information to produce an answer. Since the documents are not all physically stored on the same server, a distributed Q/A system deals with parallel tasks and load balancing. Even if some similarities with PHASER can be considered, the main difference is that a node ends its own work as it delivers the message.

Baeza-Yates et al. (1999) stated that the *Ranking problem* is fundamental in Information Retrieval. It can be solved with machine learning as summarised by Liu and others (2009). However, adopted processes usually manage many documents; this is not a realistic case of PHASER, where the rank is provided relying on little information.

3 Model

In IE the term *Intelligent* usually refers to Artificial Intelligence (AI) applied to environments, where technology offers more than static rooms as introduced by Augusto et al. (2013). In this Section we propose our method, a Pervasive Human-centred Architecture for Smart Environmental Responsiveness (PHASER): it is a distributed solution for an IE which provides a ubiquitous environment. The global intelligence is built upon single entities that show responsive behaviours and collaborate with each other to better support the user. PHASER has been firstly presented by Di Mauro et al. (2017). This Section recalls general aspects of our model: the description of what each node represents and how it constitutes a network with similar entities.

3.1 A Smart Entity

In our concept, PHASER gives a role to each entity who interacts with the others. Possible entities are *objects* and *people* interacting with those objects. We make use of an abstract concept of *node* to include the needs of both the entities. In real scenarios, objects are AI-powered devices, considered an important step on an evolutionary process that is affecting modern communication devices (Atzori et al., 2014; Lòpez et al., 2012). People, instead, are represented with their personal smartphone which acts as an interface.

Each object interacts with other connected entities, providing services and responding to questions. A graph results and objects individuate its nodes. For this reason, we will refer to objects as nodes as well.

3.2 Model of PHASER

A single node represents an entity in the environment. We formally define a node as a tuple:

$$N(\iota, Cnf_\iota, Close_\iota, Discovered_\iota, oBC_\iota)$$

where ι is a unique identifier of the node in the environment and $Close$ is a set of related nodes in the environment. $Discovered$ collects nodes connected after unforeseen interactions. $Close$ and $Discovered$ contain identifiers of the remote nodes. ι establishes connections and interacts with nodes in both the sets. Each node specifies a configuration Cnf, which determines ι's role in the environment. The configuration comprises inputs, outputs and how it reacts to network events. In details:

$$Cnf_\iota = (name_\iota, type_\iota, class_\iota, env_\iota, I_\iota, O_\iota, P_\iota)$$

where *type*, *class* and *env* classify ι according to an ontology, while *name* labels it. I and O represent inputs and outputs respectively; they divide data into channels as in Equation 1 for multimodal interaction, where c_x is a channel code and $RG_{c_x} = \{r_{i_1}, r_{i_2}, \ldots, r_{i_{c_x}}\}$ is a set of regular expressions. If N_{i_ι} and N_{o_ι} are the number of input and output channels, we define I_ι and O_ι in Equation 2.

$$Ch_j = (c_j, RG_{c_j}) \quad (1)$$

$$I_\iota/O_\iota = \bigcup_{1 \le x \le N_{i_\iota/o_\iota}} \{Ch_x\} \quad (2)$$

Input and Output compose the Business Card (BC) and it represents what a node may accept; each object exposes its own BC to the connected nodes; received BCs will be stored in oBC. The approach discussed in this work ranks $Close$ and $Discovered$ peers by obtaining confidences from their BC. PHASER nodes compose the network. There is not a hierarchic organisation, so all the nodes are at the same level. The network does not need a specific topology, but we assume that an expert of the considered domain designs it.

The presented formalism defines a PHASER node which establishes a connection towards other similar entities. This is the core part of our system: a distributed model where single peers interact with people - through I/O modules - and with others. Input and Output modules are intentionally generic because each node can have a customised the interaction. This approach aims at supporting Natural User Interfaces (Wigdor and Wixon, 2011).

The discussed formulae are the core part useful to understand the introduced improvements. A detailed description of the PHASER model has been provided by Di Mauro et al. (2017).

4 Navigation Problem

In PHASER, each node is expected to have a knowledge, circumscribed to its own domain: a fridge should understand questions about food or

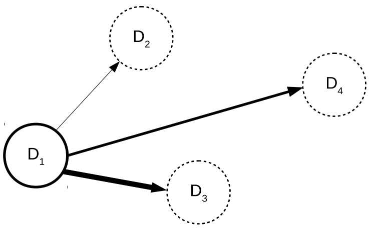

Figure 1: The thickness of each arc is proportional to the probability of $D_{2,3,4}$ to accept the received request

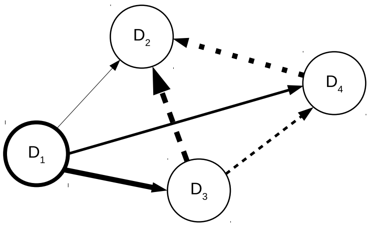

Figure 2: A command X from D_1 reaches D_2 via D_3 or D_4. Arcs' thickness is the match percentage of X; the dashed line concerns the path's length

ingredients; commands about lighting and heating are *out-of-context*. However, the environment could contain other devices to manage those requests. In such a configuration, a network of PHASER devices is populated of entities with partial knowledge, where nodes have a common strategy to propagate *out-of-context* requests.

Each node is able to interact with users. It means that each of those entities may manage commands for any other node in the network. They could handle non-matching requests in two ways: *(i)* broadcast them to all connected nodes or *(ii)* individuate the most probable nodes. The second approach is preferred in very active networks - i.e. smart museums - or where a large percentage of nodes processes unknown requests; moreover, the first approach easily overloads the network. This *"Navigation problem"* aims at forwarding the request to the best candidate; the model follows a *depth-first-search* by iterating on a sorted list of close nodes. The ranking is obtained with a greedy approach. The navigation continues until a node finds a response or all the sub-network is explored. Figure 1 reports a graphical representation of the sorting phase that is performed in the navigation presented in Figure 2. D_1 chooses the next node in the interaction by sorting the adjacent vertices.

This work analyses how the discussed probability is obtained, presenting the current technique, and investigating improvements that support partial matching to tailor the propagation on the network. The outcome is a different rank of the list of close nodes explored in the navigation of the graph.

4.1 Perfect matching

A network of PHASER nodes starts from a topology designed by experts. During the interaction, the network adapts connections to maximise the local utilities on each arc. As it is set up, each pair of connected nodes shares a business card. It comprises a set of active channels and, for each channel, a set of regular expressions (regex) for the accepted inputs.

The *"Navigation problem"* is solved by sorting the adjacent nodes according to their matching with the exposed regular expressions. Nodes with higher value of matching will be firstly called in forwarding. Inputs can be on multiple channels, so the matching complies with the structure. In this version, M calculates the value of matching as defined in Equation 3:

$$M(R,n) = \sum_{0 \leq i < |R|} m(R_i, n)/|R| \qquad (3)$$

$$m\left(R_x, n\right) = \begin{cases} 1 & \text{if } R_x \text{ is valid for } n \\ 0 & otherwise \end{cases} \qquad (4)$$

where n is the considered device, R is the request, divided into $|R|$ channels. The expression "R_x is valid for n" of the Equation 4 means that exists a regular expression of n that matches with R_x. The higher M is, the higher is the probability that n accepts R. $M(R,n) = 1$ means a perfect match.

4.2 Imperfect matching

The currently adopted approach is based on full matching where the outcome of each $m(x,n)$ is dichotomous. The calculated value M is then normalised to the size of R - involved channels in R -. This approach highly depends on the accuracy of the design of the set of regular expressions. Moreover, generic regexs - i.e. ".*" - accepts everything. This case undesired, as if a node accepts this input it will attract many requests with the consequence of not being able to process all of them; this would create a *black hole*, that uselessly overloads the network.

Alternative approaches perform linguistic analyses of the received question. The investigated solution is based on partial matching; it provides a *confidence* of the input, used to refine the ranking of the adjacent nodes. The improvement still must prefer a perfect matching, but it does not completely exclude the opposite case. Then, we propose a revised version of the formulae seen in Section 4.1 by introducing $m^v_{l_x}$ as the confidence of v on channel x and adapting M as follows:

$$M(R, n) = \prod_{0<x\leq|R|} max \left\{ \left(m^{R_x}_{i_1}, \ldots, m^{R_x}_{i_n} \right) \right\} \tag{5}$$

The function in Equation 5 supports multiple channels and a set of possible grammars for each of them, but $m^v_{l_x}$ is now the probability that the token v from the request is accepted on an input l_x. This probability can be calculated with two strategies: regex-based and *bag-of-words* (BoW). The former approach calculates the longest substring that matches on each provided regex on the proper channels; this obtained length is then normalised on the total length of the request. The *bag-of-words* method, instead, splits both the request and the stored accepted inputs in two bags of words - B_{req} and B_{input} respectively - and calculates how many words of the request match on the total set. This value is then normalised on $|B_{input}|$. Both the strategies are locally performed by nodes on received questions that must be forwarded. No global dictionaries are saved in order to maintain a scalable distributed system where each node has partial knowledge about the others.

Since any NLP approach provides a confidence of the evaluated input request, other strategies have been considered. However, these approaches present drawbacks that will be discussed in Section 6.

5 Experiments and Results

This Section reports experiments conducted to compare the three discussed approaches in PHASER: perfect matching, partial matching, and BoW. Full and partial matching methods rely on regular expressions and assess how much the request matches the provided regexs. The system has been tested by simulating a smart house with 5 networked PHASER nodes. The considered nodes are TV, Microwave Oven (M), Fridge (F), Kettle (K), Alarm clock (A).

We considered a star-like network with TV in the middle. We tested two kinds of configurations for input representation. A request is delivered to the TV, which forwards the request to the node with the highest confidence; this is obtained with the different approaches. The network is design to let *Kettle* being the winner.

Table 1 collects data where inputs are represented with a BoW style; in Table 2, instead, inputs are represented as regular expressions. Each node used OpenDial by Lison and Kennington (2016) to manage a dialogue.

command	*perfect*	*partial*	*BoW*
prepare a tea	K (1.0)	K (1.0)	K (1.0)
warm	**A (0.1)**	M (0.44)	M (0.5)
warm water	**A (0.1)**	**A (0.1)**	K (0.667)
wake me	**A (0.1)**	A (0.438)	A (0.5)

Table 1: Winner device and confidence for each request. Each node had a bag-of-word style inputs. Bold cells refer to unsuccessful evaluations

command	*perfect*	*partial*	*BoW*
prepare a tea	K (1.0)	K (1.0)	**K (0.1)**
warm	**A (0.1)**	M (0.44)	K (0.33)
warm water	**A (0.1)**	A (0.9)	K (0.667)
wake me	**A (0.1)**	A (0.778)	A (0.4)

Table 2: Winner device and confidence for each request. Each node had a regex style inputs. Bold cells refer to unsuccessful evaluations

6 Discussion

The presented process operates in a context where the current node n is not able to understand the request r and it prefers to share it with the network, refraining from broadcasting. The node n gathers a confidence on r to sort the adjacent nodes, preferring nodes with higher values. A sequence results, where the first node is the best candidate to accept the request.

Results show that a *full matching* is not always a good choice. It requires a precise design of each regex, exposing the structure of accepted inputs; moreover, this strategy does not always discriminate different nodes and fails in many cases. *Partial matching* provides finer values and nodes are better sorted. However, this approach easily creates *black holes*, nodes that attract many inputs because of a wrong design. The BoW model gave the best results with two benefits: *(i)* the network is easier to design; *(ii)* each node could share unstructured data, improving local security.

Other strategies have been investigated. We considered more refined systems based on SRGS; however, this method has been excluded for many reasons: *(i)* SRGS requires a complete grammar from adjacent nodes and this may generate security issues because they expose a detailed structure of accepted inputs; *(ii)* grammar-based methods introduce overheads compared with the adopted approaches, due to the engine needed to recognise the request on the model represented by the grammar.

7 Conclusions

This paper presented PHASER, a distributed model for Human-Computer Interaction in Intelligent Environments. This work aims at improving the *Navigation Problem*, where a node forwards a received command if it is not able to understand or process it. Since the node operates with partial knowledge about both the request and the environment, it tries to analyse the input and choose the best adjacent node.

The most crucial part is not a refined linguistic analysis of each request, but a quick confidence on how much each adjacent node could be a good candidate to understand that request. This requirement is motivated by two reasons: *(i)* this process is part of a longer step where a user is waiting for a response; *(ii)* all the evaluations rely on information each node shares with others. In order to deeply understand the command, the node should expose sensible data and it is not always desired in a distributed context.

The work focused on three strategies: perfect and partial matches with regular expressions and a bag-of-words model. This last approach has given the best results with positive aspects mainly related to easy network design and security of each node. The investigated methods are just used to rank close peers on as an *out-of-context* request reaches the current node. It operates without understanding the request, so finer considerations are not possible. The considered approaches do not limit PHASER nodes in adopting more refined techniques in assessing and categorising an input request.

References

L. Atzori, A. Iera, and G. Morabito. 2014. From "smart objects" to "social objects": The next evolutionary step of the internet of things. *IEEE Communications Magazine*, 52(1):97–105.

J. C. Augusto, V. Callaghan, D. Cook, A. Kameas, and I. Satoh. 2013. Intelligent environments: a manifesto. *Human-Centric Computing and Information Sciences*, 3(1):1–18.

R. Baeza-Yates, B. Ribeiro-Neto, et al. 1999. *Modern information retrieval*, volume 463. ACM press New York.

D. Di Mauro, J. C. Augusto, A. Origlia, and F. Cutugno. 2017. A framework for distributed interaction in intelligent environments. In *European Conference on Ambient Intelligence*, pages 136–151.

A. Hunt and S. McGlashan. 2004. Speech recognition grammar specification version 1.0. *W3C Recommendation.*

W. Li, T. Logenthiran, W. L. Woo, V. T. Phan, and D. Srinivasan. 2016. Implementation of demand side management of a smart home using multi-agent system. In *IEEE Congress on Evolutionary Computation*, pages 2028–2035.

P. Lison and C. Kennington. 2016. Opendial: A toolkit for developing spoken dialogue systems with probabilistic rules. *ACL 2016*, page 67.

T. Liu et al. 2009. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3:225–331.

T. S. Lòpez, D. Ranasinghe, M. Harrison, and D. McFarlane. 2012. Adding sense to the internet of things: An architecture framework for smart object systems. *Personal and Ubiquitous Computing*, 16(3):291–308.

S. Pajares Ferrando and E. Onaindia. 2013. Context-aware multi-agent planning in intelligent environments. *Information Sciences*, 227:22 – 42.

M. Surdeanu, D.I. Moldovan, and S.M. Harabagiu. 2002. Performance analysis of a distributed question/answering system. *IEEE Transactions on Parallel and Distributed Systems*, 13(6):579–596.

M. Valentino, A. Origlia, and F. Cutugno. 2017. Multimodal speech and gestures fusion for small groups. In *Workshop on Designing, Implementing and Evaluating Mid-Air Gestures and Speech-Based Interaction.* in press.

S. Valero, E. del Val, J. Alemany, and V. Botti. 2016. Enhancing smart-home environments using magentix2. *Journal of Applied Logic.* in press.

D. Wigdor and D. Wixon. 2011. *Brave NUI world: designing natural user interfaces for touch and gesture.* Elsevier.