Natural Language Generation in the context of Multimodal Interaction in Portuguese

José Casimiro Pereira, António J. S. Teixeira, Joaquim Sousa Pinto

***Abstract* —– In many situations, systems and people have to interact and share information. With demographic ageing a group of users that deserves special attention are the older adults. Multimodal interaction aims to improve the interaction between human users and electronic devices. A key technology for the output part of a Multimodal system is conversion of the concepts handled by systems into readable text or audible speech output, area with almost no work for Portuguese. As a starting point for the development of output generation modules for use in multimodal interaction contexts, we developed, as proof of concept, a prototype of a NLG module to support transmission of information on drugs to older adults based on a data-driven translation approach. First sample results of the generation are presented.**

Keywords: Natural Language Generation, NLG, multimodal interfaces, fission, adaptability to users, data to text, Portuguese

I. INTRODUCTION

In many situations, systems and people have to interact and share information. New devices and technologies are changing the way people use devices like cell-phones or small computers. The naturalness, efficiency and effectiveness of interaction are a major concern [1].

A. Users are not all equal

Systems and interfaces are made for people and users are not equal. It is expected that different users, with different characteristics and capabilities, have different expectations when they use a device or system. Coetzee and co-workers [2] argue that most systems do not take this into consideration, since they are prepared for, what they call, the "average user". These new interfaces need to adapt to user, processing input and generating adapted output.

With demographic ageing, probably the greatest achievement of mankind, a group of users that deserves special attention are the older adults. With suitable natural interfaces, the introduction of technological solutions can facilitate their daily life, fighting isolation and exclusion, increasing their pro-activity, work capacity and autonomy [3].

B. Multimodal interaction

Either due to environmental conditions, where this interaction is made, either due to physical or mental limitations of human users, sometimes, is necessary to use various modalities, simultaneously, to improve this interaction.

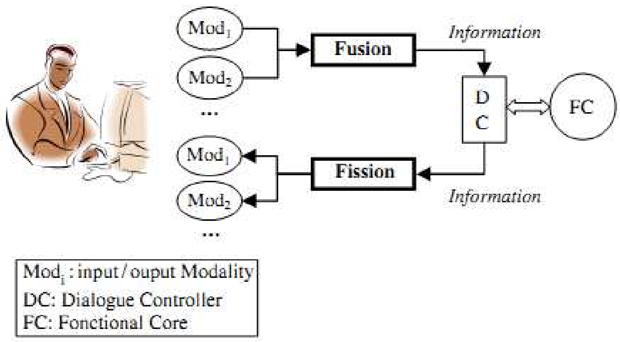


Figure 1 – Simplified multimodal system architecture (adapted from [6]).

Multimodal access aims to improve the interaction between human users and electronic devices, like computers or cell phones. This new concept makes it possible to use various forms of interaction (sound, gestures, GUI, etc.). With these types of interaction it is also possible to create an environment where a user connect transparently to the same content, regardless of the type of device used (mobile phone, PDA, computer, etc.). This interaction will be, naturally, limited by the characteristics of users and characteristics of the devices used [2, 4, 5].

Multimodal systems (as presented in Figure 1) are systems mainly composed by 3 modules, besides input and output modalities: *Fusion*, *Fission* and *Dialogue Manager*[6, 7]. Fusion deals with the perception of information that the user provides to the system. Fission, on the other hand, deals with information that the system will provide to user, in response to a user interaction. Dialogue Manager is the "brain" of multimodal systems, since it processes the data provided by fission module, determining what response should be given by system, through fission module.

C. Output inmultimodal

A key technology for the output part of a Multimodal system is conversion of the concepts handled by the Interaction, Dialog Manager or even the application into readable text or audible speech output.

For humans, speaking or writing are natural ways of interacting. Since man invented computers, several efforts were made to provide computers with the ability of produce "human" language as good as humans can do [8].

One of these efforts are Natural Language Generation (NLG) systems. This kind of systems are defined by Ehud Reiter as systems that "*generate texts in English and other human languages*" [9]. NLG systems are requested to decide "how" to say (or write) after deciding "what" to say [9-11]. This means that NLG systems should be able to behave as a human being, producing text or speech that is syntactically and semantically correct, in addition to being correctly contextualized.

D. Portuguese and lack of work for our language

It is estimated at about 240 million the number of people who speak the Portuguese language worldwide [12, 13]. This makes the Portuguese language the 5th most spoken language in the world. It is expected that this number grows up to 335 million by 2050 [14]. This reality brings new opportunities in the study of multimodal interfaces, with especial concern about Portuguese natural language generation. A survey in the publications of the scientific community showed that there are very few work carried out on the Portuguese language, related with multimodal systems and NLG. Most of it (if not all) is about Brazilian Portuguese.

Our aim is to produce a system, which interacts with a user, mainly with multimodal interface, in Portuguese language. This system should be able to recognize the kind of user that is interacting with it, and the context of that interaction, and produce written, oral or visual messages in European Portuguese.

It is our intension that our system is developed in the area of health care. At this moment, we are working in producing, on what we call MedicalCare NLG system. This first prototype intends to help elderly people to identify and take their medication.

This paper is organized as follows:

Section 2 presents a brief introduction to Natural Language Generation (NLG) area, to contextualise this paper; *Section 3* describes the user modelling, since it is important to know and understand the characteristics of users of our systems; *Section 4* describes the prototype of our first NLG module; *Section 5* presents the results obtained with the prototype presented in section 4, and on *last section*, we conclude, discussing future work.

II. NATURAL LANGUAGEGENERATION – A BRIEF OVERVIEW

Bateman and Zock[11] argue that NLG systems need to map some non-linguistic source (such as database data) to some linguistic form of speech. Nevertheless, this is not an easy task, since it involves the necessity of make several difficult choices. First, it determines the motivation of the conversation. Secondly, it has to decide what should be communicated. After, how the message is going to be expressed, and lastly, the message it transmitted to user.

Those choices are usually made by a "dialog manager" [10, 15].

A. General architectureandphases

Reiter and Dale presented in their book "Building Natural Language Generation Systems" [16, 17] the prototypical architecture of application-oriented NLG system (see **Figure 2**))and their major processing phases: *Document Planning*, *Microplanning,* and *Surface Realization*:

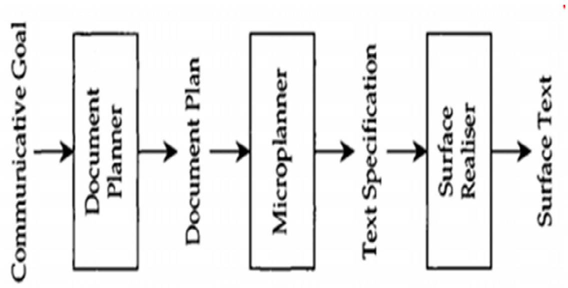


Figure 2– An NLG system architecture (adapted from [16])

- **Document Planning** – Document Planning is the first phase of NLG Systems. It decides what information is used for communication. In most systems is the most important phase. It includes a Content Determination module and Document Structuring module. Content Determination decides what messages should be used in final utterance. Document Structuring organizes messages selected previously, making coherent phrases and paragraphs.

- **Microplanning** – The second phase of NLG systems materializes 'document plan' produced in previous phase.

It includes three modules: Lexicalization – choice of the words needed to relay the information contained in document plan; Aggregation – decides how much information should be used; Referring Expression Generation – decides what phrases should be used to identify entities or objects to the user.

Surface Realization – last phase. Previous phase prepare the text to be generated, but do not generate it. That is the work made here. In Surface Realization messages are prepared to be presented to user.

Next figure presents an example of work done by a surface realization engine. This example was produced by Novais, et al. [18], and uses, in `a)`, simply a template to produce the desired output; in `b)`, some extra information to format template output; and in `c)`, a more complex structure, where some linguistic knowledge, such as person, mode, verb, verb tense or adverbial modifiers are used to configure the original template.

	Input	Expected output
(a)	template #17	[You]$_{agent}$ [should deliver]$_{action}$ [your results]$_{patient}$ on Friday.
(b)	template #17, patient=essay, action=not_complete	[You]$_{agent}$ [did not complete]$_{action}$ [your essay]$_{patient}$ on Friday.
(c)	template #17, agent=teacher, determiner=possess, action=give, tense=future, patient=talk, determiner=indefinite	[Our teacher]$_{agent}$ [will give]$_{action}$ [a talk]$_{patient}$ on Friday.

Figure 3– An NLG systemarchitecture (adaptedfrom[16])

B. Classes ofApproaches

Lemon, in [10], outlined the three main approaches to generating system utterances in dialogue systems: *template-basedNLG*, *conventionalNLG* and, more recently, *trainable* generation.

•**Templates** – Templates use pre-build text forms. It is mainly used in industrial dialogue systems. They are present in state-of-the-art systems, too. Templates are not necessary easy to build and they have the disadvantage of having little flexibility, since for each situation is necessary to rewrite them. However, generation mechanism do not need substantial modifications [19]. Output speech quality could be as good as others kinds of NLG systems;

•**Conventional NLG** – has been presented by Reiter and Dale in [16].. As already presented, it has three main modules: a *text planner*, a *sentence planner* and a *surface realiser*;

•**Trainable NLG** – modules that use *trainable NLG* are more complex than those described above. Here, different strategies are used. Several use ontologies as main structure of archiving corpora. These ontologies are then used to produce new utterances. Systems as SPaRKy[20, 21], ELEON [22] or BabyTalk[23, 24] are examples of these approach. Others, like Mountain [25, 26], use aligned corpora and tools like MOSES [27] to statistically produce new utterances.

NLG Systems Tools and Resources

In past years, several systems and tools have been developed in the field of NLG. Here, we present the most relevant for our objectives.

2) Systems

Representative systems are:

- *Mountain*

Mountain [25, 26] was designed by Langner and A. Black. Its operating principle is simple but not trivial. Mountain acts like a machine translation system. It uses two parallel-aligned languages. One is called the internal language. The second is called the target language. Internal language describes the several states the system has. The target language corresponds with the messages to be sent, in natural language, to the end user.

Mountain uses the MOSES translation machine [27] as the kernel tool.

Mountain's work is done in four steps: First, it is defined a parallel corpus with the internal and target language. Secondly, Mountain is trained with that corpus. The goal is to train a translation model, which is capable of translating from the structured internal language to appropriate natural language. Third, the target language (the natural language from corpus) is used to train the language model, which is responsible for the quality of output utterances. Fourth, after language model training, MOSES is used by Mountain to generate output utterances, related with "sentences" from the internal language.

- *SINotas*

Natural Language Generation in Portuguese is still a subject largely unexplored. Novais, Araújo and co-workers developed SINotas[28, 29] which is a system primarily developed as a testbed for NLG research in the

(Brazilian) Portuguese language. SINotas was developed at University of São Paulo, Brazil.

It is an aligned data-to-text [30] system. SINotas use parallel corpus. This corpus consists of two hundreds and a half aligned data-to-text sentences. It takes the students' grades (the raw data), from academic records, and correspond it to an utterance which explains the situation of that student, comparing his grade with his progress over the semester. It compares, too, student's grade with all class' grades.

Novais, Araujo and co-workers argue that SINotas is useful for students and professors, since it allows, both of them, assess students' performance, in a friendly way.

- *BabyTalk*

BabyTalk[23, 24] is one of the most recent projects of Ehud Reiter. It is also a data-to-text project.

This system aims to help doctors and nurses, working in a Neonatal Intensive Care Unit (NICU), make better decisions about the health of newborn sick babies. The amount of available data is, frequently, too big, and these professionals need analyse it, most often with time pressure. Data refer to patient's reports, clinical examinations, measurements of biometric sensors and so forth. Since time to interpret it is often scarce, BabyTalk aims to produce textual reports, with that data, allowing minimizing medical errors.

BabyTalk project has two main objectives: *BT-Nurse* and *BT-Doc*. BT-Nurse intends to summarise available data of a 12-hour shift, in order to inform an incoming nurse of what had happen before. BT-Doc summarises, too, available data in a 6-12 hours shift. In this case, these reports are addressed for junior doctors to provide more support for their decisions.

As corollary, BabyTalk has two other secondary objectives: *BT-45* and *BT-Family*. BT-45 summarizes the last 45min of available data. BT-Family will generate daily reports for parents of NICU babies in order to inform and reassure them.

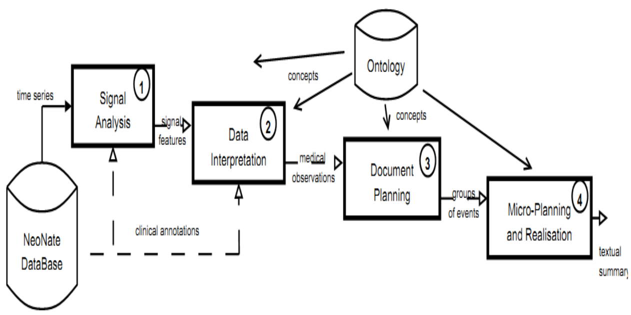


Figure 4 – BT-45 architecture
(adaptedfrom [24])

Figure 4 shows the architecture of BT-45, which was the first module built. Textual summaries are generated in four stages, all of which access a domain ontology which includes information about NICU concepts.

First stage is *Signal Analysis* (1) which extracts the information provided by medical devices; *Data Interpretation* (2) analyses the provided data by first stage and infer actions to perform or report; *Document Planning* (3) selects the most important events from earlier stages and groups them into a tree of linked events. Finally, *Microplanning and Realisation* (4) translates this tree into coherent text [23].

2) Tools

- *Moses*

MOSES [27, 31] is a statistical machine translation system. With it is possible to train a model to translate parallel corpus. This parallel corpus is a collection of two strictly aligned corpora files. Both files should have de same number of lines, and for each line, in the source file, the text it contains is the translation of the same line in the target language file.

MOSES work is done in conjunction with GIZA++ [32] and with SRILM [33]. The purpose of these two tools is to provide a correct alignment between the two sentence-aligned files with corpus (GIZA++), and for building and applying statistical language models (SRILM).

Moses features include two types of translation models: *phrase-based* and *tree-based*. In *phrase-based*, Moses only uses two files.One for describing the alignment between words from the source file and target file. This file is the main file for this model, since the decoder consult it to figure out how to translate the input text file, from the source input language to the target language. The other is the configuration file [34].

The *tree-based* mode is slightly different from `phase-based` model. In phase-based, exists a direct and atomic translation from the input phrase to the output phrase. In tree-based, the mapping rules use grammar rules, which include variables. These models are called tree-based, because the translation is organized like a tree.

Another relevant feature about MOSES is its ability to handle with ambiguous inputs. When a machine translation system is designed to process only one input hypothesis, if two or more inputs occur, an error could happen. Moses is capable of processing inputs relatively ambiguous. These inputs are stored in the form of lattices. The processing of these lattices will produce the best possible translation [34].

- *SimpleNLG*

SimpleNLG[35, 36] is a Java API toolkit, developed under the supervision of Ehud Reiter at the University of Aberdeen – Scotland, which does basic NLG lexicalization and realization. SimpleNLG is an ongoing project. It is intended to function as a "realization engine" for Natural Language Generation architectures [35]. Basically, it provides interfaces offering direct control over the realization process. By realization process, Gatt and Reiter [36] mean the way phrases are built and combined. Despite SimpleNLG is specially designed for English language, it was adapted to French and German with satisfactory results.

SimpleNLG operates with three modules [35]: *Lexicon/ morphology system* – decides what words should be used; *Realiser* – generates texts from a syntactic form; and, *Microplanning* – realizes simple sentence aggregation.

It is specially designed for data-to-text applications, which generate texts, written in natural language, having by source numerical or other non-text data. SimpleNLG has been used successfully in several projects. One of the latest was Baby Talk.

III. USERMODELLING

How to adapt to users? When a system is produced, it is designed to be used by users. Therefore, it is important to understand and predict users' preferences and expectations, in order to develop good interfaces.

Besides the study of users' preferences, how to acquire user personality information is another issue in current investigation. How to adapt system behaviour to personality and mood of users? Mairesse and Walker [37, 38] proposed a system named Personage. This tool aims to adapt linguistic style of utterances produced, based on analysis of users' politeness, personality or linguistic style.

Another important work was made by Coetzee, Viviers and Barnard [2]. They proposed a model to determine the best possible combination of input and output modalities. This model, designated Cost Model, is a mathematical tool that takes in consideration the user´s profile and preferences.

By user profile, the Cost Model refers to users' capabilities. It is argued that every individual has different abilities (what he can do) and several disabilities (what he cannot do) to deal with systems interface. It is emphasized the abilities rather than the disabilities.

Table 1 presents a list of assumptions that is possible to make about user's abilities linked with output. Our work is especially interested in the abilities marked with the dashed line.

Table I
LIST OF ASSUMPTIONS ABOUT USER'S ABILITIES LINKED WITH OUTPUT
(adapted from [2])

Ability associated with output	Assumptions
Can See	None
Can Hear	None
Can Read	User can see
Can Read (simplified text)	User can see
Can Understand South African Sign Language	User can see
Can Feel	None
Can Understand Braille	User can feel
Can Lip Read	User can see

User's abilities are not the only responsible for constraints over interaction with devices. Other factors can interfere. Table 2 presents a list of users' individual preferences, when interacting with systems. These preferences are directly related with their senses (sight, hear, touch). These user preferences might, as well, be influenced by users' literacy.

Table II
USER'S PERCEPTUAL PREFERENCE
(adapted from [2])

Perceptual Preference	Description
Visual (V)	Individual prefers pictures, graphs and diagrams.
Aural (A)	Individual prefers spoken words
Read/Write (R)	Individual prefers reading and writing texts
Kinaesthetic (K)	Individual prefers to move his/her body and manipulate things with his/her own hands

IV. TOWARDS A FIRST NLG MODULE – NLG FOR A MEDICINE REMINDER IN PORTUGUESE

We studied several tools, and none provides a simple methodology for developing NLG systems. All systems were built to solve a particular problem, and none of them could be extended to suit our needs.

This motivated us to build our own system. We had focused in elderly people, because they have special needs, especially if we realize that their memory and physical capabilities are not how they used to be, and, it is well known that Portuguese people are getting old. The aim of this first prototype NLG module, that we called MedicalCare NLG, is to take that in consideration.

A. Scenario

The scenario, which we have designed, plans to address the situation where a person is taking medicines. The system should interact with the user in is natural language (in this case, Portuguese), and assist him/her on their needs.Figure 5 shows a hypothetical scenario, where a user is interacting with MedicalCare system.We intend, for instance, that if a person asks what time the medicine is to be taken, system should reply telling him/her what he/she wants and complementing with the characteristics of the medicine (shape, colours, quantity, etc.).

To do that, this NLG module uses three types of users: the person who really is going to use the system; the person who configures the system (it could be a technician or a relative) and the system itself.

Figure 5 – Hypotheticalscenario,with a user is interacting with MedicalCare system (createdusingfrom`secondlife.com`)

B. TheAdoptedApproach: MachineTranslation NLG

Our aim was to translate information retrieved from database system into natural language. We have analysed several systems and tools. Most of them were template-based. Every one we tried had several drawbacks that made impossible to use them.

One of last works analysed was Langner's work [25, 26] based on translation and training from a corpus. We decided to use a similar approach,

C. Some ImplementationDetails

The MedicalCare NLG system is still in a very embryonic stage. As Langner[25], we have decided to use MOSES [27], GIZA++ [32] and SRILM [33] tools.

This system consists in three main components:

- The *DBMS* (Data Base) component: database software responsible for archiving all kind of data. The characteristics of users, their medical prescription, etc.;
- the*MOSES NLG machine*: module responsible for translate the data send by NLG module into Portuguese text;
- *NLG module*: the kernel of this system. It is responsible for receiving user's inputs; save/retrieve data to/from DBMS, send messages written in internal language to MOSES module, and, finally, to process NLG messages sent by MOSES, and inform user about what he/she needs to do. This interaction will be made by text, sound (voice) and/or images.

Next figure details the proposed architecture of the NLG module.

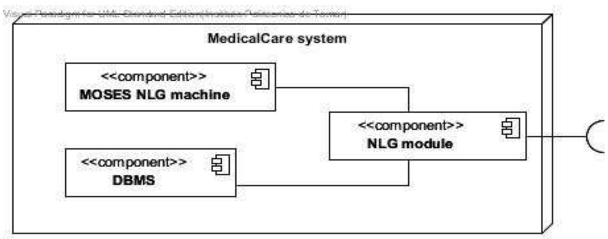


Figure 6 – The proposed architecture

D. Corpus

The aligned parallel corpus is needed in order to obtain the mapping between the internal "language" and the natural utterances that we would like to teach.

Our collected corpus is still a very experimental corpus. It consists of about 284 sentences. In fact, it is a collection of two parallel-aligned data sets: the internal language and the target language.

As explained before, MOSES uses two parallel-aligned languages. The internal language, that reflects the database data, consists from three to seven ordered tokens. First token: patient name, second: medicine name, third: medicine dosage, fourth: medicine type, fifty: medicine main colour, sixth: medicine secondary colour, and seventh: the dosing frequency. It can grow, since there are more data types in the database.

The target language consists of Portuguese sentences, describing the data expressed in the internal language.

To test, we made a second version, resulting of some processing. All tokens of the internal language were converted to numbers, and the number of tokens was established in five. The numbers we used here are the primary key of the corresponding record in database. Therefore, the meaning of tokens is as follows. First token: patient' ID, second: medicine ID, third: medicine dosage. Fourth token combines three data types: medicine type ID, medicine main colour ID, medicine secondary colour ID. Fifth and last token, the dosing frequency. A null value (zero) express that there is no value for that token.

Next tables present some examples of the training corpus. They represent the same data with the reported changes made to the internal language.

Table III
Part of the First Aligned Parallel Corpus

Internal language	**Corresponding target language**
Carla Costa CCCC	Sra. Costa não se esqueça de aplicar o medicamento CCCC
Carla Costa CCCC 1 spray incolor	Aplica o spray incolor, do medicamento CCCC
Mário Silva GGGG 2 comprimido verde alface 8	Deve tomar 2 comprimidos verde alface, do medicamento GGGG, de 8 em 8 horas
José Godinho AAAA 4 cápsula azul branco 6	O Sr. José Godinho vai tomar 4 cápsulas azuis e brancas, de 6 em 6 horas, do medicamento AAAA
João Pereira HHHH pomada branco	Sr. Pereira deve aplicar a pomada do medicamento AAAA

Table IV
PART OF THE SECOND (PROCESSED) VERSION OF THE PARALLEL CORPUS

Internal language	**Corresponding target language**
8 3 0 000 0	Sra. Costa não se esqueça de aplicar o medicamento CCCC
8 3 1 610 0	Aplica o spray incolor, do medicamento CCCC
2 7 2 160 8	Deve tomar 2 comprimidos verde alface, do medicamento GGGG, de 8 em 8 horas
9 1 4 283 6	O Sr. José Godinho vai tomar 4 cápsulas azuis e brancas, de 6 em 6 horas, do medicamento AAAA
1 8 3 300 0	Sr. Pereira deve aplicar a pomada do medicamento AAAA

E. TrainingProcess

The next figure represents the sequence used to train and use Moses.

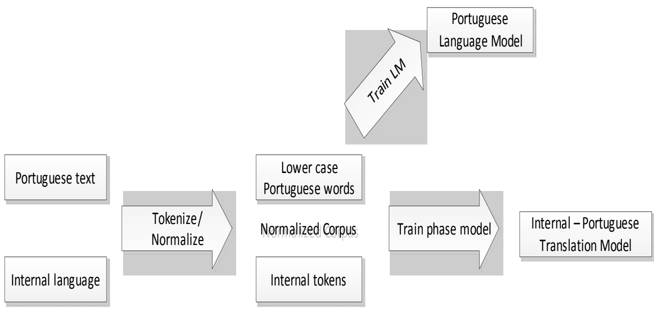


Figure 7 – The training process (adapted from [7])

Training the models for MOSES required several steps. First, the corpus was tokenized. In this step, every special symbol (like a dot, comma, semi-colon, etc.) presented in corpus need to be separated from regular words, inserting space characters before and after it. Because MOSES scripts are not prepared to work with Portuguese words, it was done manually.

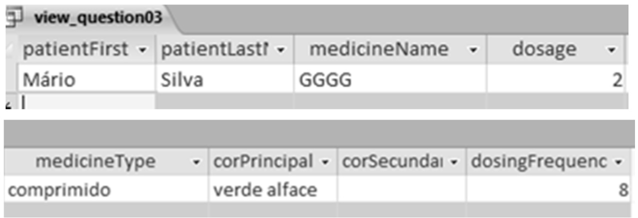
Second, both internal and target language were converted to lowercase. Third, we used SRILM tool to build the language model for the target language. Due to small corpus it was only possible to build a 2-gram language model.

Finally, this prepared corpus with the language model was used in the training of MOSES.

V. FIRSTRESULTS

As said before, our aim was to translate raw data into Natural Text. We defined a plausible question, inspired by MedicalCare philosophy: *What medicine should I take now?*, and tested the system's answer.

For the first version of the system, presented in this paper, MedicalCare NLG system only has the database and the translation machine. Therefore, all other work is done manually. To answer a question, a view was made, resulting in a table with the necessary data from the database to drive the generation process.


view_question03

patientFirst	patientLastN	medicineName	dosage
Mário	Silva	GGGG	2

medicineType	corPrincipal	corSecundai	dosingFrequenc
comprimido	verde alface		8

Figure 8 – the extracted data for this example

The resulting data, which corresponds to the description of the medicine to be taken, was transformed to a list, as, for example:*Mário Silva GGGG 2 comprimidoverdealface 8* (in English: Mário Silva GGGG 2 green-lettuce pills), with space characters separating each field. This list has correspondence in the internal language.

Afterthat, wetranslatethelistwithMOSES (Figure 9), andtheresult, for ourexample,was: *não se esqueça de tomar 2 comprimidos verde alface,, do medicamento de 8 em 8 horas* (inEnglish: do notforget to take 2 green-lettucepills, of medicine in 8 in 8 hours).

Secondly, a new set of experiences were made using the processed version of the corpus. The raw list was converted into numbers, matching each field with their corresponding primary key. The result for our example was: `2 7 2 160 8`. As for this first version of the system we opted to not have real medicine names, we used names generic names as "aaaa" or "bbbb". Inclusion of real names is anticipated as straightforward.

The result of the generation process (Figue 10) wasslightly better: *não se esqueça de tomar 2 comprimidosverdealface , do medicamento**gggg** , de 8 em 8 horas* (in English: do not forget to take 2 green-lettuce pills, of medicine gggg, in 8 in 8 hours).On first experience, Moses did not return medicine name ("gggg"), which occurred in second set of experiences.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we argued on the need and usefulness of NLG systems and how they can benefit people, especially elderly or very young ones. We also present briefly NLG structure and report about systems and tools that are useful to support development in this area.

Because the population of Portugal is getting old, we are especially interested in studying tools to this kind of users, hoping to contribute to improve their daily live. Our first experiments seem to show that is possible to produce NLG systems with Portuguese language by adopting a translation approach trained using existing tools and using an adequate parallel corpus.

The results we obtained are encouraging, although our work is still in a very initial phase. In the near future, we expect to have evaluation results, an improved corpus, the first developments in user modelling and application to new scenarios.

Our work intends to also contemplatethe use of multimodality, with special focus in output. We plan to use text, audio, and, where possible, visual information in 2D or 3D. The use of *avatars* is a strong possibility.

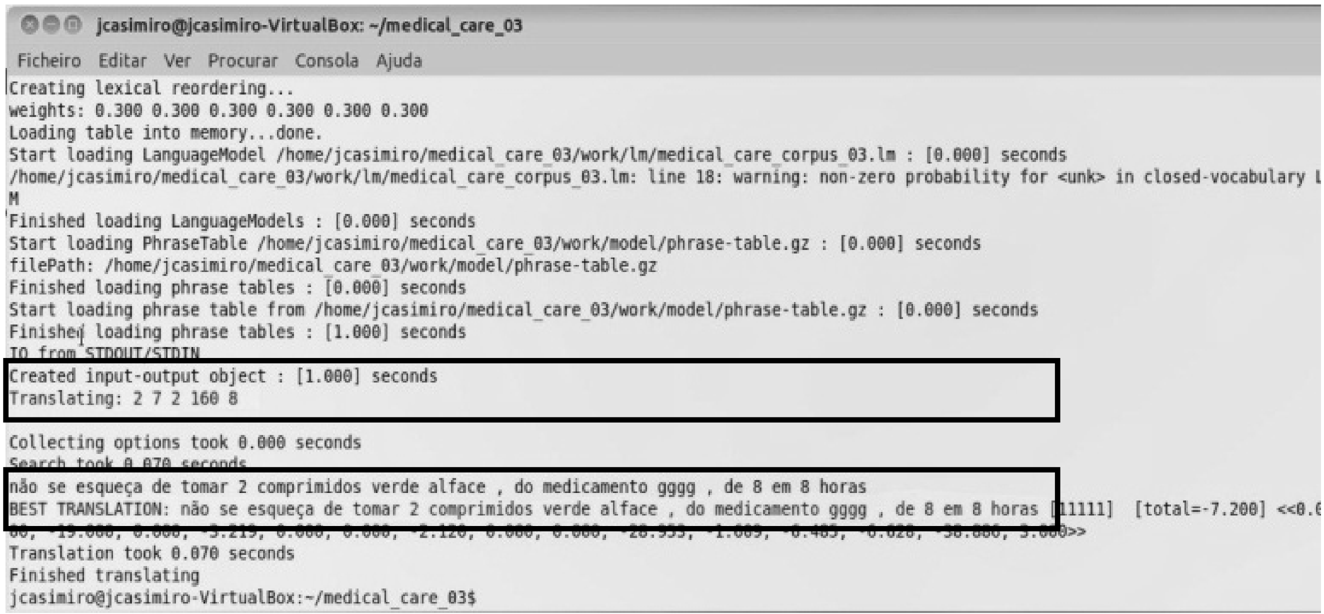


Figure 9 – Second MOSES translation

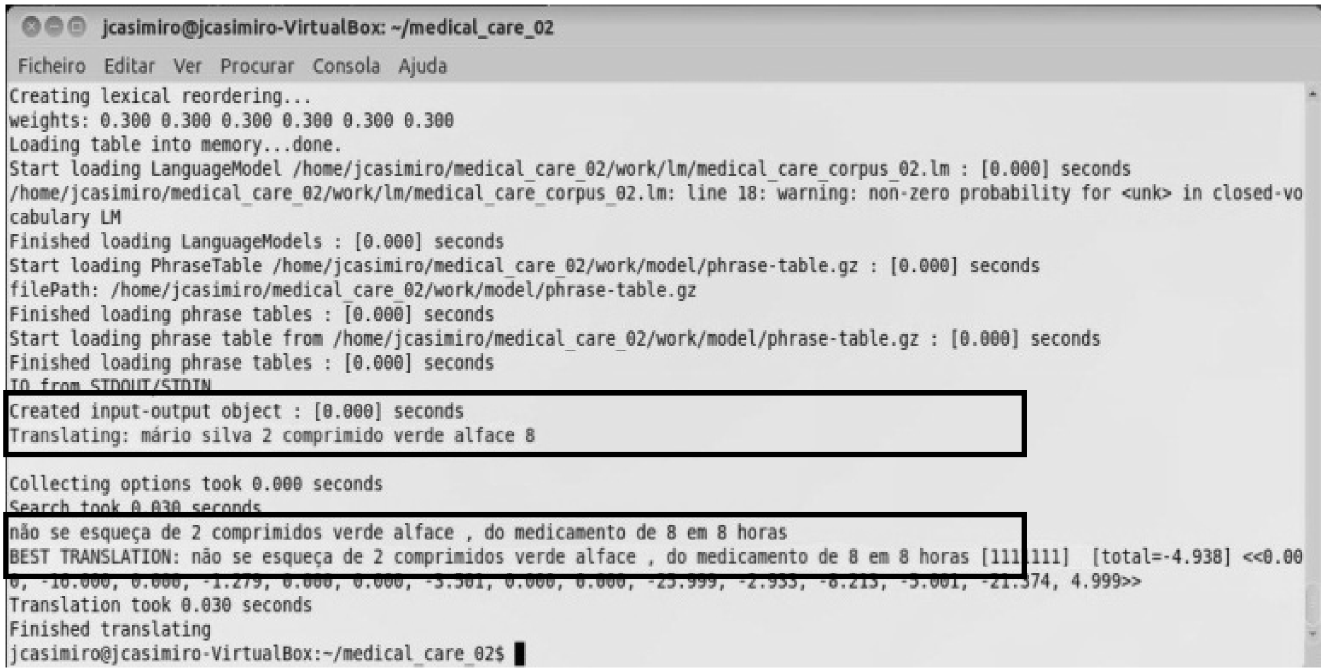


Figure 10 – First MOSES translation

REFERENCES

[1] N. Sebe, “Multimodal interfaces: Challenges and perspectives,” *J. Ambient Intell. Smart Environ.*, vol. 1, no. 1, pp. 23-30, 2009.

[2] L. Coetzee, I. Viviers, and E. Barnard, “Model based estimation for multi-modal user interface component selection,” in 20th Annual Symposium of the Pattern Recognition Association of South Africa (PRASA 2009), South Africa, 2009, pp. 1-6.

[3] Living Usability Lab. "Living Usability Lab Project," 2010/June/12, 2010; http://www.livinglab.pt/.

[4] W3C. "Multimodal Access (W3C)," 2010/March/26, 2010; http://www.w3.org/standards/webofdevices/multimodal.

[5] N. O. Bernsen, "Multimodality Theory," *Multimodal User Interfaces - From Signals to Interaction* Signals and Commmunication Technologies, D. Tzovaras, ed., pp. 5-29: Springer Berlin Heidelberg, 2007.

[6] C. Rousseau, Y. Bellik, F. Vernier *et al.*, “A framework for the intelligent multimodal presentation of information,” *Signal Process.*, vol. 86, no. 12, pp. 3696-3713, 2006.

[7] B. Dumas, D. Lalanne, and S. Oviatt, "Multimodal Interfaces: A Survey of Principles, Models and Frameworks," *Human Machine Interaction*, Lecture Notes in Computer Science D. Lalanne and J. Kohlas, eds., pp. 3-26, Berlin/Heidelberg: Springer-Verlag, Berlin/Heidelberg, 2009.

[8] D. Jurafsky, and J. H. Martin, *SPEECH and LANGUAGE PROCESSING, 2nd ed.*, 2nd. ed., New Jersey - USA: Pearson - Prentice Hall, 2009.

[9] E. Reiter, "Natural Language Generation," *The Handbook of Computational Linguistics and Natural Language Processing*, A. Clark, C. Fox and S. Lappin, eds., pp. 574-598, USA: Wiley-Blackwell, 2010.

[10] O. Lemon, “Learning what to say and how to say it: joint optimization of spoken dialogue management and Natural Language Generation,” *Computer Speech and Language*, vol. 25, pp. 210-221, 2010.

[11] J. Bateman, and M. Zock, "Natural Language Generation," *The Oxford Handbook of Computational Linguistics*, Ruslan Mitkov (Eds.). ed., pp. 284-304, Oxford, UK: Oxford University Press, 2004.

[12] Ministério Negócios Estrangeiros de Portugal. "Instituto Camões de Portugal," 2010/June/21, 2010; http://www.instituto-camoes.pt/accao-do-instituto-camoes-no-ambito-da-lingua/p%C3%A1gina-3.html.

[13] L. V. Alves. "Português é terceira língua mais falada no mundo," http://www.tvciencia.pt/tvcnot/pagnot/tvcnot03.asp?codpub=26&codnot=8.

[14] A. Lusa. "Falantes de Português irão aumentar para 335 milhões em 2050 "; http://www.publico.pt/Cultura/falantes-de-portugues-irao-aumentar-para-335-milhoes-em-2050_1429372.

[15] D. Bohus, and A. I. Rudnicky, “The RavenClaw dialog management framework: Architecture and systems,” *Comput. Speech Lang.*, vol. 23, no. 3, pp. 332-361, 2009.

[16] E. Reiter, and R. Dale, *Building Natural Language Generation Systems*: Cambridge University Press, 2000.

[17] H. Horacek, "Building Natural Language Generation Systems (book review)," *Digital Archive of Research Papers in Computational Linguistics*, 27, http://www.aclweb.org/anthology-new/J/J01/J01-2008.pdf, [2010/June/20, 2001].

[18] E. M. d. Novais, T. D. Tadeu, and I. Paraboni, “Text generation for Brazilian Portuguese: the surface realization task,” in Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas, Los Angeles, California, USA, 2010, pp. 125-131.

[19] K. V. Deemter, E. Krahmer, and M. Theune, “Real versus Template-Based Natural Language Generation: A False Opposition?,” *Computational Linguistics*, vol. 31, no. 1, pp. 15-24, 2005.

[20] A. Stent, and M. Molina, “Evaluating automatic extraction of rules for sentence plan construction,” in Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue, London, United Kingdom, 2009, pp. 290-297.

[21] A. Stent, R. Prasad, and M. Walker, “Trainable sentence planning for complex information presentation in spoken dialog systems,” in Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 79.

[22] Software and Knowledge Engineering Laboratory. "ELEON Ontology Authoring and Enrichment Tool " 2011/July/08, 2011; http://users.iit.demokritos.gr/~eleon/ELEONDescription.html.

[23] J. Hunter, E. Reiter, Y. Sripada *et al.*"BabyTalk," 2011/July/07, 2011; http://www.abdn.ac.uk/ncs/computing/research/nlg/projects/current/babytalk/.

[24] F. Portet, E. Reiter, A. Gatt *et al.*, “Automatic generation of textual summaries from neonatal intensive care data,” *Artifitial Intelligence*, vol. 173, no. 7-8, pp. 789-816, 2009.

[25] B. Langner, “Data-driven Natural Language Generation: Making Machines Talk Like Humans Using Natural Corpora,” Language Technologies Institute, School of Computer Science - Carnegie Mellon University, Pittsburgh, PA 15213, USA, 2010.

[26] B. Langner, and A. W. Black, “MOUNTAIN: A Translation-based Approach to Natural Language Generation for Dialog Systems,” in First International Workshop on Spoken Dialogue Systems Techology (IWSDS 2009), Irsee, Germany, 2009.

[27] P. Koehn, H. Hoang, A. Birch *et al.*, “Moses: Open source toolkit for statistical machine translation,” in 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, Prague, Czech Republic, 2007, pp. 177–180.

[28] R. P. A. d. Araujo, R. L. d. Oliveira, E. M. d. Novais *et al.*, “SINotas: the Evaluation of a NLG Application,” in Seventh International Conference on Language Resources and Evaluation, Valletta, Malta, 2010.

[29] E. M. d. Novais, R. L. d. Oliveira, D. B. Pereira *et al.*, “A Testbed for Portuguese Natural Language Generation,” in Seventh Brazilian Symposium in Information and Human Language Technology, Sao Carlos, Sao Paulo, Brazil, 2009.

[30] E. Reiter, “An architecture for data-to-text systems,” in Proceedings of the Eleventh European Workshop on Natural Language Generation, Germany, 2007, pp. 97-104.

[31] Moses. "Moses - statistical machine translation system," 2011/June/27; http://www.statmt.org/moses/.

[32] F. J. Och, and H. Ney, “A systematic comparison of various statistical alignment models,” *Journal Computational Linguistics*, vol. 29, no. 1, pp. 19-51, 2003.

[33] Speech Technology and Research Laboratory. "SRILM - The SRI Language Modeling Toolkit," 2011/July/02, 2011; http://www.speech.sri.com/projects/srilm/.

[34] Moses. "Moses manual," 2011/June/08; http://www.statmt.org/moses/manual/manual.pdf.

[35] E. Reiter. "simplenlg - Java API for Natural Language Generation," 2011/July/07, 2011; http://code.google.com/p/simplenlg/.

[36] A. Gatt, and E. Reiter, "SimpleNLG: a realisation engine for practical applications," in Proceedings of the 12th European Workshop on Natural Language Generation, Athens, Greece, 2009, pp. 90-93.

[37] F. Mairesse, and M. A. Walker, "Towards personality-based user adaptation: psychologically informed stylistic language generation," *User Modeling and User-Adapted Interaction,* vol. 20, no. 3, pp. 227-278, 2010.

[38] F. Mairesse, and M. A. Walker, "Controlling User Perceptions of Linguistic Style: Trainable Generation of Personality Traits," *Computational Linguistics,* vol. 37, no. 3, pp. 1-50, 2011.