Proceedings

Developing Open-Source Roguelike Games for Visually-Impaired Players by Using Low-Complexity NLP Techniques †

Luís Fernández-Núñez, Darío Penas, Jorge Viteri, Carlos Gómez-Rodríguez [ID] **and Jesús Vilares *** [ID]

CITIC, LyS Research Group, Department of Computer Science & Information Technologies,
Facultade de Informática, Campus de Elviña, Universidade da Coruña, 15071 A Coruña, Spain;
fernandezn.luis@gmail.com (L.F.-N.); dariosoyyo@gmail.com (D.P.); j.viteri.letamendia@udc.es (J.V.);
carlos.gomez@udc.es (C.G.-R.)

* Correspondence: jesus.vilares@udc.es; Tel.: +34-881-01-1364

† Presented at the 3rd XoveTIC Conference, A Coruña, Spain, 8–9 October 2020.

check for updates

Abstract: The prominent graphic component of video games greatly limits the accessibility of this type of entertainment by visually impaired users. We make here an overview of the first games developed within an initiative for the development of roguelike games adapted to visually impaired players by using Natural Language Processing techniques. Our approach consists of integrating a multilingual module that automatically generates text descriptions of what is happening within the game. The user can then read such descriptions by means of a screen reader.

1. Introduction

The offer of games accessible to users with severe visual impairments such as blindness is still limited both in genres and titles. This is mainly due to the prominent role played by graphics in games, thus limiting their accessibility. However, it may also happen when the graphic aspect of the game is secondary, as in the case of *roguelike* games. In these games, the player controls a hero who explores a dungeon complex fighting monsters, avoiding traps and discovering treasures. This genre has a great number of fans, both players and developers, who pay little attention to the graphic aspect on behalf of the game mechanics and the player experience. Actually, ASCII-based graphics are not rare even at present [1]. Nevertheless, little attention has been paid to their accessibility by blind players.

For several years, our research group has been offering to our students the chance of developing especially adapted open-source roguelike games as their Final Year Projects. These games should be accessible to both sighted and visually impaired players by applying low-complexity *Natural Language Processing* (NLP) techniques. To do this, these games offer, apart from the standard game mode, a so-called *descriptive mode* intended for visually impaired users. In this extra game mode, the classic graphic representation of the dungeon and its elements is replaced by natural language descriptions automatically generated by a multilingual module, which can be read using screen-reading software.

The system should be flexible and easily extensible and modifiable by third parties. Thus, other members of the game community could improve aspects of the basic game or add new languages to the description module. Our intention is that any person with basic programming skills and high-school linguistic skills should be able to extend the system, at least in part, to a new language. Additionally, our experience with low-resource languages suggested that the number and complexity

of the linguistic resources to be used should be low. Therefore, they should be simple to obtain or build, if needed, since the free availability of these types of resources is often limited in many languages [2].

2. System Architecture

Figure 1 shows the general architecture of our games. The main difference with respect to regular roguelikes is the addition of a so-named *Text Description Engine* module. This new component takes as input the (current) world model and, together with the information provided by the game logic in response to the commands of the user, it generates a textual description of the dungeon and what is happening within the game. Such description is generated according to the linguistic resources available and the language selected. This new module follows the classic architecture of a Natural Language Generation (NLG) system, consisting of three stages [3]:

- *Content planning*. Determines the content and structure of the description.
- *Microplanning*. Selects the words and syntactic structure to be used to express such content.
- *Surface realisation*. Integrates all this information and transforms the abstract representation of the message into actual text to be presented to the user.

Seeking expressivity and variety, as stated before, especial attention has been paid to the microplanner. If the user is presented with monotonous descriptions that are repeated in the same terms again and again, he will soon get tired. To avoid this situation, we take advantage of the so-called *linguistic variation* (a.k.a. *linguistic flexibility*): the ability of our languages to express the same message in very different ways (and vice versa) [4].
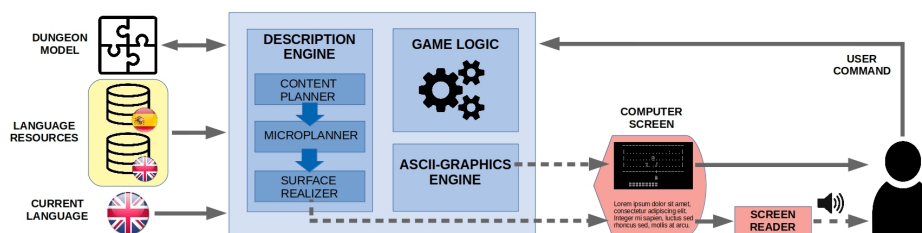

Figure 1. General architecture of the system.

3. Linguistic Processing

The techniques applied when implementing the generation module involve different levels of linguistic processing, from the lexical to the pragmatic level, without losing sight of multilingualism.

At the *lexical and morphological levels*, it is necessary to provide the system with a *vocabulary* containing every element, action and situation that may occur or appear within the game. For every term, this dictionary also includes its corresponding morphosyntactic information (e.g., word category, part-of-speech tag, etc.) together with its inflectional variants due to changes in gender, number, person, etc. Moreover, every term must be conveniently indexed with an external identifier through which it will be referenced in those program entities or processes involving it. For each language available in the game, a separate vocabulary must be created, with all the translations of a given term sharing the same external identifier. These dictionaries were implemented as JSON files because of their simplicity and flexibility.

Regarding the *syntactic level*, once the description module has retrieved those vocabulary terms corresponding to a gameplay event, they are arranged according to a *generation grammar* to form a meaningful message describing the event. This grammar is structured into *subgrammars* according to the syntactic structures generated (e.g., noun phrases) and the different contexts in which they are used (e.g., combat). Again, subgrammars and rules are indexed using an external identifier to link them with the events they describe, independently of the specific language being used at a given time.

These grammars are defined using context-free rules and kept in the form of JSON files using a format inspired in feature structure-based grammars [5].

With respect to the *semantic Level*, we have experimented with an alternative vocabulary structure where terms are not selected and organized individually, but grouped forming *synsets* (synonym sets) instead. Again, an external identifier has been used to identify and link the entries (i.e., synsets) between languages.

Finally, at the *discourse and pragmatic levels*, some features were successfully tested. One of them consisted on changing the adjectives used to describe a character taking into account its current state (e.g., wounded or hungry), thus introducing a subjective point of view. Another feature was persistence over time; for example, after having combat in a room, subsequent descriptions of that room should reflect damages and signs of a fight.

4. Results

Until now, three games were developed within our initiative for developing open-source roguelike games accessible for visually impaired users by using low-complexity NLP techniques:

1. *The Inner Eye*, by Luis Fernández-Núñez;
2. *The Accessible Dungeon*, by Darío Penas;
3. and *Hamsun's Amulet*, by Jorge Viteri.

all af them available at our website. At this point, the languages that were addressed are: English, Spanish, Galician and Dutch. Preliminary work has been made for Japanese, too. Further details about the solutions applied for the implementation of these games can be found in our previous work [6].

From a social point of view, our proposal favors integration, since both sighted and visually impaired users can play the same game and, thus, share common experiences. By helping draw students' attention to accessibility concerns, students will be aware of them when they go on to become software developers. Finally, from an academic point of view, this type of final-year projects allow us to introduce the student to NLP in a practical and engaging way.

References

1. Craddock, D.L. *Dungeon Hacks: How Nethack, Angband, and other Roguelikes Changed the Course of Video Games*; Press Start Press: Canton, OH, USA, 2015.
2. Rehm, G.; Uszkoreit, H. (Eds.) *META-NET White Paper Series*; Springer: Berlin, Germany, 2011. Available online: http://www.meta-net.eu/whitepapers (accessed on 18 August 2020).
3. Reiter, E.; Dale, R. *Building Natural Language Generation Systems*; Cambridge University Press: Cambridge, UK, 2000.
4. Arampatzis, A.; van der Weide, T.P.; van Bommel, P.; Koster, C. Linguistically-motivated Information Retrieval. In *Encyclopedia of Library and Information Science*; Marcel Dekker: New York, NY, USA, 2000; Volume 69, pp. 201–222.
5. Carpenter, B. *The Logic of Typed Feature Structures*; Number 32 in Cambridge Tracts in Theoretical Computer Science; Cambridge University Press: Cambridge, UK, 1992.

6.   Vilares, J.; Gómez-Rodríguez, C.; Fernández-Núñez, L.; Penas, D.; Viteri, J. Bringing Roguelikes to Visually-Impaired Players by Using NLP. In Proceedings of the LREC 2020 Workshop Games and Natural Language Processing, Marseille, France, 11 May 2020; Lukin, S.M., Ed.; European Language Resources Association (ELRA): Paris, France, 2020; pp. 59–67. Available online: https://lrec2020.lrec-conf.org/media/proceedings/Workshops/Books/Games-NLPbook.pdf (accessed on 18 August 2020).