BAD: An assistant tool for making verses in Basque

Manex Agirrezabal, Iñaki Alegria, Bertol Arrieta
University of the Basque Country (UPV/EHU)
maguirrezaba008@ikasle.ehu.es, i.alegria@ehu.es, bertol@ehu.es

Mans Hulden
Ikerbasque (Basque Science Foundation)
mhulden@email.arizona.edu

Abstract

We present work on a verse-composition assistant for composing, checking correctness of, and singing traditional Basque *bertsoak*—impromptu verses on particular themes. A performing *bertsolari*—a verse singer in the Basque Country—must adhere to strict rules that dictate the format and content of the verses sung. To help the aspiring *bertsolari*, we provide a tool that includes a web interface that is able to analyze, correct, provide suggestions and synonyms, and tentatively also sing (using text-to-speech synthesis) verses composed by the user.

1   Introduction

In the Basque Country there exists a long-standing live performance tradition of improvising verses—a type of ex tempore composition and singing called *bertsolaritza*. Verses in *bertsolaritza* can be seen as discourses with strict rules governing the technical structure of them: verses must contain a certain number of lines and each line must have a defined number of syllables, certain lines have to rhyme in certain patterns, and so forth.

In this paper we present a web-based assistant tool for constructing verses (*bertsoak*) according to the rules of *bertsolaritza* (Garzia et al, 2001).

If the reader is interested in this topic, we recommend watching the 2011 film *Bertsolari*[1] [2], directed by Asier Altuna.

---

[1] IMDB: http://www.imdb.com/title/tt2058583
[2] Trailer on: http://vimeo.com/9355066

2   Relationship to earlier work

There exist some prior works dealing with Basque verse-making and computer technologies, such as BertsolariXa (Arrieta et al., 2001), which is a rhyme search tool implemented as finite-state automata using the two-level morphology formalism. The tool also contains other features, including semantic categorization of words, narrowing word-searches to certain themes, etc. While BertsolariXa focuses mostly on the word-level, the current work also includes constraints on overall verse structure in its implementation as well as a synonym search tool, a melody suggestion system, and possibilities for plugging in text-to-speech synthesis of verses.

2.1   The *Bertsolari* tradition

Bertsolaritza is very ingrained in the Basque Country and championships, competitions and get-togethers on *bertsolaritza* are quite common. Usually the competitors in such event, called *bertsolaris*, are given a theme to produce a verse on under some very limited time constraints.

But the Basque Country is not the only place that hosts such troubadour traditions—similar customs are present in many other countries such as Cuba, Brazil, Argentina, etc. The goal of the current tool is to be generalizable, and so applicable to various strategies of verse improvisation, and possibly be useful not only for Basque speakers, but also for others.

Below we briefly present an example of a verse made in the Basque Country. In 1986 Andoni Egaña (a well-known *bertsolari*) was asked to sing a *bertso* and assigned a topic. In the verse, he was asked to play the role of an old person who lived alone, and who realized that he could

not even tie his shoes. Within a few seconds he composed and sang three verses. Here, we analyze the first verse.

<div align="center">

Verse:

Gazte aroan ibili arren
gustora tirriki-tarra,
denbora honen joan etorriak
ederki jo dit gitarra,
gorputza daukat ximeldurikan
ta eskuen punta zaharra,
denborarekin seko galdu det
gazte aroko indarra,
ez al da pena gizon mardul bat
hola ibili beharra.

Translation:

Even when I was young
I was always on a spree
over time
I have been punished
I have a crumpled body
and the tip of the hands very old,
Over time I lost
the strength I had when I was young,
It's a shame that a strong man
has to end up like me.

</div>

The special charm of *bertsolaritza* improvisation is that people proficient in the art can quickly express a variety of ideas, although they are working with very restrictive rules concerning the number of syllables in words they use, and how the words must rhyme. We must take into account that Andoni Egaña was able to sing this verse within a few seconds of being given the topic, and also, that it complies exactly with a certain metric. In this case, the verse contains eight lines, each odd line consisting of ten syllables, and each even line of eight syllables, with the even lines rhyming.

Formal training in the *bertsolari* tradition also exists in the Basque Country. In the last 20 to 30 years, an important movement has developed that aims to provide instruction to upcoming generations on how to create verses (orally or in writing). This kind of instruction usually takes place in learning centers called *bertso-eskolak*, which in English roughly means, "verse-making schools." The proliferation of this movement has produced a strong base of young *bertsolaris*, of whom many achieve an outstanding level of improvisation skills.

3 The BAD tool

BAD is the acronym for "*Bertsotarako Arbel Digitala*", roughly "Digital verse board." The aim of the tool is to serve as a general assistant for *bertsolari*-style verse composition and help verse-making learners in their learning process.

This tool has been developed using the PHP programming language, but it contains certain parts developed using finite-state technology. The main functions of this tool, which will be discussed in more detail in the next five sections, are the following: visualization of the verse structure, structure checking, rhyme and synonym searching and verse singing.

3.1 Verse structure

The main rules of the *bertsolari* verse are that a verse must consist of a certain predefined number of lines and each line in turn, of a predefined number of syllables. Traditionally, about a hundred different schemes are used, and the tool provides support for all these patterns. For example, the structure called "*Hamarreko handia*" has ten lines and ten syllables in the odd-numbered lines, and eight syllables in the even-numbered lines. In this structure, the even-numbered lines have to rhyme. Selecting this scheme, the tool will mark the corresponding lines with their requirements.

The web interface can be seen in figure 1, which shows the general layout of the tool, illustrated with the example verse referred to above—we see that each line has been approved in terms of line length and syllable structure by the tool.

We have designed a database in which the main verse structures are saved so that when the user selects one verse schema, the system knows exactly the number of lines it must contain, where must it rhyme and how many syllables each line should have. Those schemata are also linked to melodies, each melody corresponding to one possible structure.

3.2 Structure checking

After writing the verse, the system can evaluate if it is technically correct, i.e. if the overall structure is correct and if each line in the form abides by the required syllable count and rhyming scheme. The syllable counter is implemented using the *foma* software (Hulden, 2009), and the implementation (Hulden, 2006) can be found on the homepage of

Figure 1: A verse written in the BAD web application.

foma.[3]

Separately, we have also developed a rhyme checker, which extracts special patterns in the lines that must rhyme and checks their conformity.

These patterns are extracted using *foma* (see section 3.4) after which some phonological rules are applied. For example, an example rule *era* → {era, eda, ega, eba}, models the fact that any word ending in *era*, for example, *etxera*, will rhyme with all words that end in *era*, *eda*, *eba* or *ega*. These rhyming patterns have been extracted according to the phonological laws described in (Amuriza, 1981).

3.3 Synonym search

Usually, people who write verses tend to quickly exhaust their vocabulary and ideas with to express what they want to say, or encounter problems with the number of syllables in various tentative words they have in mind. For example, if the verse-maker wants to say something containing the word "family," (*familia* in Euskera, a four-syllable word) but is forced to use a three-syllable word in a particular context, the interface provides for possibilities to look for three-syllable synonyms of the word *familia*, producing the word *sendia*— a word whose meaning is otherwise the same, and made up of three syllables.

For developing the synonym search, we used a modified version of the Basque Wordnet (Pociello

---

[3] http://foma.googlecode.com

et al., 2010), originally developed by the IXA group at the University of the Basque Country. Within Wordnet we search the synsets for the incoming word, and the words that correspond to those synsets are returned.

3.4 Rhyme search

The classical and most well-known problem in *bertsolaritza* concern the rhyming patterns. As mentioned, various lines within a verse are required to rhyme, according to certain predefined schemata. To search for words that rhyme with other words in a verse, the BAD tool contains a rhyme search engine. In the interface, this is located in the right part of the BAD tool main view, as seen in figure 2.

The rhyme searcher is built upon finite-state technology, commonly used for developing morphological and phonological analyzers, and calls upon the freely available *foma*-tool, to calculate matching and nonmatching rhyme schemes.

Its grammar is made up of regular expressions that are used to identify phonological patterns in final syllables in the input word. The result of the search is the intersection of these patterns and all the words generated from a morphological description of Basque (Alegria et al., 1996)—that is, a list of all words that match both the required phonological constraints given (rhyming) and a morphological description of Basque.

Based upon figure 2, if we search rhymes for the word *landa* (cottage), the system proposes a

Figure 2: The response of the rhyme search engine.

set of words that can be filtered depending on the number of syllables required. Among this list of words, we can find some words that end in *anda*, such as, *Irlanda* (Ireland) or *eztanda* (explosion), but through the application of phonological equivalency rules we also find terms like *ganga* (vault).

3.5 Singing synthesis

Another characteristic, as mentioned, is that, in the end, the verses are intended to be sung instead of only being textually represented. Based on other ongoing work in singing synthesis, we have designed a system for singing the verses entered into the system in Basque.

This is based on the "singing mode" of the Festival text-to-speech system (Taylor et al., 1998). The advantage of using this is that Festival is open-source and has given us ample opportunities to modify its behavior. However, as Festival does not currently support Basque directly, we have relied on the Spanish support of the Festival system.[4]

Based on current work by the Aholab research team in Bilbao—a lab that works on Basque speech synthesis and recognition—we have implemented a singing module for BAD, based on the text-to-speech HTS engine (Erro et al., 2010). Our application is able to sing the composed verses entered into the system in Basque, with a choice of various standard melodies for *bertsolaritza*.[5]

4 Discussion and future work

Now that the BAD tool has been developed, our intention is to evaluate it. To make a qualitative evaluation we have gotten in touch with some verse-making schools (*bertso-eskola*), so that they can test the system and send us their feedback using a form. Once the evaluation is made, we will improve it according to the feedback and the system will be made public.

Our ultimate goal is to develop a system able to create verses automatically. To achieve this long-term goal, there is plenty of work to do and basic research to be done. We have in our hands a good corpus of 3,500 Basque verse transcriptions, so we intend to study these verses from a morphological, syntactical, semantical and pragmatic point of view.

In the short term, we also plan to expand the synonym search to be able to provide searches for semantically related words and subjects (and not just synonyms), like hypernyms or hyponyms. The Basque WordNet provides a good opportunity for this, as one is easily able to traverse the WordNet to encounter words with varying degrees of semantic similarity.

Another feature that we want to develop is a system that receives as input a verse together with a MIDI file, and where the system automatically sings the verse to the music provided.

Finally, in order for the system to be able to provide better proposals for the verse artist—including perhaps humorous and creative proposals—we intend to work with approaches to computational creativity. We are considering different approaches to this topic, such as in the work on *Hahacronym* (Stock et al., 2005) or the *Standup* riddle builder (Ritchie et al., 2001).

Figure 3: The BAD application before entering a verse, showing two possible rhyme patterns.

Acknowledgments

References

Iñaki Alegria, Xabier Artola, Kepa Sarasola and Miriam Urkia, "Automatic morphological analysis of Basque", Literary and Linguistic Computing, ALLC, 1996

Xabier Amuriza, "Hiztegi errimatua", Alfabetatze Euskalduntze Koordinakundea, 1981

Bertol Arrieta, Iñaki Alegria, Xabier Arregi, "An assistant tool for Verse-Making in Basque based on Two-Level Morphology", 2001

Daniel Erro, Iñaki Sainz, Ibon Saratxaga, Eva Navas, Inma Hernáez, "MFCC+F0 Extraction and Waveform Reconstruction using HNM: Preliminary Results in an HMM-based Synthesizer", 2010

Joxerra Garzia, Jon Sarasua, and Andoni Egaña, "The art of bertsolaritza: improvised Basque verse singing", Bertsolari liburuak, 2001

John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introducción a la teoría de Autómatas, Lenguajes y Computación", Pearson educación, 2002

Mans Hulden, "Foma: a finite-state compiler and library", Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session, p. 29–32, 2009

Mans Hulden, "Finite-state syllabification", Finite-State Methods and Natural Language Processing, p. 86 – 96, Springer, 2006

Kimmo Koskenniemi. "Two-level morphology: A general computational model for word-form recognition and production". Publication 11, University of Helsinki, Department of General Linguistics, Helsinki, 1983.

Eli Pociello, Eneko Agirre, Izaskun Aldezabal, "Methodology and construction of the Basque WordNet", 2010

Graeme Ritchie, "Current directions in computational humour", Artificial Intelligence Review, Volume 16, Number 2, p. 119 – 135, Springer, 2001

Graeme Ritchie, Ruli Manurung, Helen Pain, Annalu Waller, Dave O'Mara, "The STANDUP interactive riddle builder", Volume 2, Number 2, p. 67 – 69, IEEE Intelligent Systems, 2006

Oliviero Stock and Carlo Strapparava, "Hahacronym: A computational humor system", Proceedings of the ACL 2005 on Interactive poster and demonstration sessions, p. 113 – 116, Association for Computational Linguistics, 2005

Paul Taylor, Alan W. Black and Richard Caley, "The architecture of the Festival speech synthesis system", International Speech Communication Association, 1998