# Humorist Bot: Bringing Computational Humour in a Chat-Bot System

Agnese Augello, Gaetano Saccone, Salvatore Gaglio
*DINFO - University of Palermo*
*Viale delle Scienze Ed.6*
*90128 Palermo, Italy*
*{augello,saccone}@csai.unipa.it, gaglio@unipa.it*

Giovanni Pilato
*ICAR-CNR*
*Viale delle Scienze Ed.11*
*90128 Palermo, Italy*
*g.pilato@icar.cnr.it*

## Abstract

*A conversational agent, capable to have a "sense of humour" is presented. The agent can both generate humorous sentences and recognize humoristic expressions introduced by the user during the dialogue. HumoristBot makes use of well founded techniques of computational humor and it has been implemented using the ALICE framework embedded into an Yahoo! Messenger client. It includes also an avatar that changes the face expression according to humoristic content of the dialogue.*

## 1. Introduction

The interest about enhancing the interface usability of applications and entertainment platforms has increased in last years. The research in human-computer interaction on conversational agents, named also chat-bots, and natural language dialogue systems equipped with audio-video interfaces has grown as well. One of the most pursued goals is to enhance the realness of interaction of this kind of systems.

One of the most important feature in conversations between human beings is the capability to generate and understand humor. As reported in [1] "Humor is part of everyday social interaction between humans", and omputational humour [3] deals with the automatic generation and recognition of humor.

In last years verbally expressed humour has been analyzed in literature[2], concerning in particular very short expressions (jokes). A one-liner is a short sentence with comic effects, simple syntax, intentional use of rhetoric characteristics (e.g., alliteration, rhyme), and frequent use of creative language constructions[6][17].

Since during a conversation the user says short sentences, one-liners, jokes or gags can be good candidates for the generation of humorous sentences. As a consequence, computational humour techniques regarding one-liners can be customized for the design of a humoristic conversational agent.

In this paper we propose an humoristic conversational agent capable to generate humoristic expressions, proposing to the user riddles, telling jokes, and ironically answering to the user. Besides, the chat-bot can detect, during the conversation with the user, the presence of humoristic expressions, listening and judging jokes and react changing the visual expression of the avatar, according to the perceived level of humor. The chat-bot shows a smiling face if the user sentence is perceived as "funny" or a neutral face if none of the features characterizing a humoristic phrase is recognized in the sentence.

The system has been implemented as a Yahoo instant messaging contact, in order to make it publicly available.

The subsequent sections report the background about chat-bots and computational humor; in section 3 the system description is given, in section 4 the Yahoo! Messenger Avatar is illustrated and in section 5 a set of experimental trials obtained on a set of humoristic and non humoristic sentences is reported. Conclusions are given at the end of the paper.

## 2. Chat-bots

Chat-bot systems are dialogue software agents able to handle a conversation with the user by means of simple matching rules. An example is the Open Source chat-bot Alice[5]. The Alice dialogue is based on a pattern matching algorithm which looks for a match between the user's sentences and the information stored in the chat-bot knowledge base. The Alice knowledge base is described by a set of question-answer modules, called categories. The categories are

0-7695-3109-1/08 $25.00 © 2008 IEEE
DOI 10.1109/CISIS.2008.117

structured with an XML-like language called AIML (Artificial Intelligence Mark-up Language). The main elements of the chat-bot knowledge base are described by means of specific AIML tags. Figure 1 shows an example of AIML category. The tag pattern encloses the user question, while the tag template encloses the chat-bot answer. The presence in the pattern of  special symbols called wildcards allows a chat-bot to obtain a partial matching between the user question and the pattern.  The template can contain other AIML tags, which enhance the dialogue chat-bot capabilities. As an example set and get tags allow chat-bot to save and get the values of variables, the system tag enables the execution of other programs,  the srai tag recursively calls the pattern matching on another category.

```
<category>
   <pattern> HELLO
   </pattern>
   <template>
     Hi, user!!!!
   </template>
</category>
```

**Fig. 1: An example of AIML category**

## 3. Computational humour

Humour is one of the most fascinating and complicated side of the human behaviour. Dealing with the cognitive aspect of humour, it contributes in getting and keeping people's attention, while it helps also retaining in memory [6].

Computational humour deals with the analysis of the humour with the main aim of computationally managing verbal humour. Two main issues of this field are the automatic generation  of humorous text and the automatic recognition of humorous sentences[6][17].

The first issue has been coped with the HAHACRONYM project whose aim was the achievement of an acronym ironic re-analyzer and generator in a given context. The humorous effect has been obtained by automatically changing some word present in an acronym preserving the rhyme and rhythm[6]. In [2], a punning riddles generation system has been proposed equipped with an interactive user interface. Other systems are illustrated in [7][8][9].

The second issue, i.e. the automatic humour recognition, is more difficult to manage since it is based on both syntax and semantics of sentences entered by the user. An attempt has been proposed in [10] where the humorous intent inside short dialogues is recognized. In another work a humour classification system over a corpus of Italian quotations manually extracted and tagged from the Wikiquote project has been implemented [11].

One of the most known attempt has been presented in [4] where the problem of recognition of verbal humour has been managed both as a classification problem and as a task of style-features recognition. In particular the attention has been focused on the type of humour present in very short sentences, called one-liners. In particular the authors have detected the main texts humoristic features analyzing the humour literature[3][12][13] and selecting the features which can be computationally detected, such as alliteration, antinomy and adult slang.

## 4. The humorist bot

Humorist bot is a chat-bot provided with sense of humor, it is capable of telling humorous anecdotes to the user and it is also capable of listening jokes, trying to understand their humorous level. The chat-bot reacts accordingly to the user jokes, showing itself smiling if it considers the sentence "funny", indifferent if it does not perceive any humour in the joke or angry if it considers the joke in poor taste.

Figure 2 shows the system architecture.

The core of the system is given by the chat-bot knowledge base, composed of three kind of AIML categories:

1. the set of standard Alice categories, which allow the chat-bot to hold a general conversation with the user;
2. a set of categories aimed at the humorist sentences generation. These categories allow the chat-bot to answer the user in a humoristic way, by means of jokes.
3. a set of categories which allow the chat-bot to recognize an humoristic intent in the user sentences. This feature is obtained connecting the chat-bot knowledge base to external resources, such as the lexical dictionary WordNet[14] and the CMU pronunciation dictionary[15], in order to detect the presence of humoristic linguistic features in the sentence.

Besides, the chat-bot has also been implemented as a Yahoo Messenger instant messaging client, in order to make the system easily accessible to the final user.

The next section describes the implemented humour recognition techniques, and explains how these techniques have been applied to the conversational agent.

Finally some example of created Aiml categories and a set of experimental results are shown.

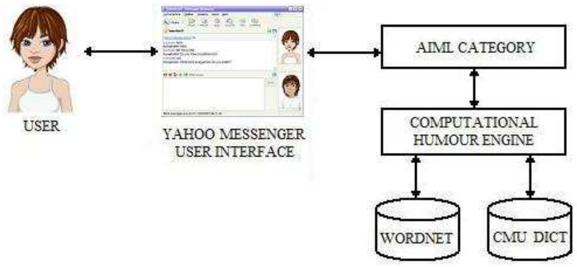

**Fig. 2 Humorist Bot Architecture**

## 4.1 Recognition of humoristic texts features

In the proposed work, we have paid more attention to the humour recognition techniques, because they are more difficult to handle but have more impact to the final user. The humour recognition has been implemented by means of the recognition, into the user sentences, of the peculiar humoristic texts features. We have considered as humoristic features those ones suggested by Mihalcea and Strapparava[4] among the features described in literature, which are the alliteration, the antinomy and the adult slang.

**4.1.1 Alliteration.** The alliteration, the rhetoric figure consisting in the repetition of a letter, a syllable or a phonetic sound in successive words is a feature present in the most of humoristic sentences. The phonetic effect induced by these repetitions captures the attention of people listening it, and in the most of cases, produces a funny effect[4].

In this work a module to recognize this feature in a sentence has been implemented. The sentence is initially processed in order to remove punctuation marks and stopwords, then its phonetic transcription is obtained using the CMU dictionary[15]. The module then analyzes the phonetic transcription of the sentence in order to detect a possible repetition of the initial phonemes in subsequent words.

The module finds an alliteration if:

- at least three words have in common the first phoneme;
- at least three words have in common the first two phonemes;
- at least three words have in common the first three phonemes.

Two examples of one-liners including one or more alliteration chains with their respective phonetic transcription are reported:

"C program run. C program crash. C programmer quit"

S IY1 . P R OW1 G R AE2 M . R AH1 N . S IY1 . P R OW1 G R AE2 M . K R AE1 SH . S IY1 . P R OW1 G R AE2 M ER0 . K W IH1 T .

"If you can't convince them, confuse them"

IH1 F . Y UW1 . K AE1 N T . K AH0 N V IH1 N S . DH EH1 M . K AH0 N F Y UW1 Z . DH EH1 M .

**4.1.2 Antinomy.** The presence of antinomies in a sentence is another feature producing an humoristic effect. A specific module for the detection of antinomies (of nouns, verbs, adverbs and adjectives) in a sentence has been developed; the module exploits the semantic relation defined into the lexical dictionary WordNet. In particular three different procedures have been implemented in order to detect into a sentence:

- a direct antinomy relation among nouns, verbs, adverbs and adjectives;
- an extended antinomy relation, which is an antinomy relation between a word and a synonym of its antonym. The relation is restricted to the adjectives;
- an indirect antinomy relation, which i san antinomy relation between a word and an antonym of its synonym. The relation is restricted to the adjectives.

These humoristic sentences contain antinomy relation:

"A *day* without sunshine is like, *night*"

"Artificial *intelligence* usually beats real *stupidity*"

"*Good* girls are *bad* girls that never get caught"

**4.1.3 Adult slang.** In many cases, the comic effect of jokes is due to the presence of words pertaining to the sexual domain, mainly used in the adult slang.

The chat-bot analyzes the presence of this kind of words using a list of terms classified as adult slang. As an example the following sentences are reported:

"100,000 sperm and you were the fastest?"

"Sex is like air; it's not important unless you aren't getting any"

## 4.2 Chat-bot knowledge base

**4.2.1 Standard AIML categories of the Alice chat-bot.** The Humoristic chat-bot knowledge base includes the set of standard Alice categories, which allow the chat-bot to entertain the user dialoguing about non humoristic topics.

In addiction to the standard Alice categories, also other AIML categories have been specifically created for the Humoristic chat-bot.

The following categories are two examples of AIML question-answer modules needed to hold this dialogue. In these categories the potentiality of the AIML tags are exploited in order to allow the chat-bot keeping trace of the conversation (*that* tag) and to set and get dialogue information (*set* and *bot* tags in the example).

*User:* What is your name?
*FunnyBot:* My name is FunnyBot.. and you? What is your name?
*User:* Marco
*Funny Bot:* Nice to meet you, Marco !!

```
<category>
<pattern>WHAT IS YOUR NAME</pattern>
<template> My name is <bot name/>.. and you? What is your name? </template>
</category>

<category>
<pattern>*</pattern>
<that>MY NAME IS FUNNYBOT... AND YOU? WHAT IS YOUR NAME?</that>
<template> Nice to meet you, <set name=”user”> <star/> </set>!! </template>
</category>
```

**4.2.2 Aiml Categories for the Generation of Humoristic Sentence.** The generation of humoristic sentences is obtained through the writing of appropriate AIML categories, exploiting the pattern matching mechanism of Alice chat-bot.

The following is an example of an humoristic dialogue:

*User:* Tell me a joke
*FunnyBot:* Do you have any preference?
*User:* Yes
*FunnyBot:* What kind of argument do you prefer?
*User:* What kind of argument do you know?
*FunnyBot:* Blondes, Lawyer, Male, Political, Religious, Computers,…
*User:* Computers.
*FunnyBot:* COFFEE.EXE is missing. Insert cup and press any key.

The previous dialogue can be obtained by means of the following categories:

```
<category>
<pattern>TELL ME A JOKE</pattern>
<template>Do you have any preference?</template>
</category>

<category>
<pattern>YES</pattern>
<that>DO YOU HAVE ANY PREFERENCE </that>
<template>What kind of argument do you prefer?</template>
</category>

<category>
<pattern>WHAT KIND OF ARGUMENT DO YOU KNOW</pattern>
<template>Blondes, Lawyer, Male, Political, Religious, Computers,…</template>
</category>

<category>
<pattern>COMPUTERS</pattern>
<template>
<think> <set name="topic">COMPUTERS</set> </think>
<random><li> Computers will never replace books. You can't stand on a floppy disk to reach a high shelf</li>
<li>Robots will be able to buy happiness, but in condensed chip form</li>
<li>COFFEE.EXE is missing. Insert cup and press any key </li>
</random> </template>
</category>
```

**4.2.3 AIML categories for humour recognition**. The recognition of humoristic sentences is obtained by means of special AIML categories which allow the chat-bot to execute the modules developed in order to detect the humoristic features previously analyzed. The modules are executed by means of the *system* tag inserted into the template, as shown in the following categories:

```
<category>
<pattern>CAN I TELL YOU A JOKE</pattern>
<template>Yes you can</template>
</category>

<category>
<pattern>*</pattern>
<that>YES YOU CAN</that>
<template><system>java HumourLevel <star/>
</system> </template>
</category>
```

This is a dialogue example that shows the chat-bot reaction to a funny joke:

*User:* Doctor gave a man six months to live. The man couldn't pay his bill, so he gave him another six months.

*FunnyBot:* You are very funny!! Good.. your joke is fantastic!!

This other example shows the chat-bot reaction to a not humoristic user sentence:

*User:* The book is on the table.
*FunnyBot:* I don't enjoy myself.. !! Your joke is stupid!!

This last example shows the chat-bot reaction to a rude joke:

*User:* 100,000 sperm and you were the fastest?
*Funny Bot:* You are a pig !! You can't tell this to a lady!!

## 5. Yahoo! Messenger

Humorist bot was connect to an instant messaging program to make it more interesting and accessible to end-users. The chat-bot is able to communicate in real time with other users and it can be incorporated into a list of contacts as a real-user.

Among the possible instant messaging programs, it has been decided to connect the conversational agent with Yahoo Messenger, since it is one of the most widely used in the world and it owns a Java classes (JYMSG) library, which allows us to interact with it. Therefore a yahoo client (Funnybot07@yahoo.it) for our agent has been created. Through the appropriate configuration of the AIML interpreter (Program D) it has been possible to realize the connection between the agent and instant messaging software.

The use of Instant Messaging program has also allowed the use of available avatars and their different expressions associated according to the different moods.

.

**Fig. 3 Possible expressions for a Yahoo Messenger avatar**

## 6. Experimental results

In order to verify the validity of the developed project, it was decided to make experiments for evaluated the classification accuracy.

So, according to what proposed in [17] it was created a humorous data set consisting of 100 positive examples (humorous phrases) extracted by specific internet sites devoted to the subject, and a non-humorous data set consisting of 100 negative examples (non-humorous phrases) always extracted from the Internet and choices between titles of newspapers, proverbs and sets definitions.

Figure 4 shows the featuring of positive data set, while Figure 5 shows that one relative to the negative data set.

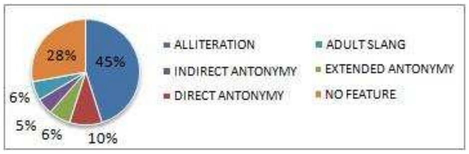

**Fig. 4 Features of the Positive Data-Set**

In a first set of experiments, we have evaluated literature algorithms implemented on each one of both data sets, in order to recognize, if present, the stylistic humour-specific features: alliteration, antinomy, and adult slang. Results report 66% of correct recognition and 34% of no recognition.

In the second set of experiment, we evaluated the number of sentences properly recognized on each one of the two data sets. The results are reported to be 81% of correct recognition and 19% of no recognition.

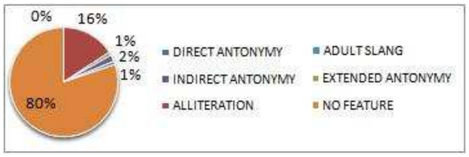

**Fig. 5 Features of the Negative Data-Set**

From the obtained experimental results, we can state that on the basis of 100 random humorous phrases, the system is able to recognize and correctly classify 66% of the sentences, while on a random set of 100 non-humorous phrases there are only 19 false positives or 19 phrases that despite being labelled as "non-humorous" are labelled "humorous" instead.

The results can be considered satisfactory, because in a real dialogue, the conversational is not made only about humorous phrases.

Then we performed a new set of experiment using a data set of 200 phrases, equally distributed between humorous and non-humorous sentences, getting more satisfactory results, given by 73% of correct recognition and 27% of no recognition.

## 7. Conclusions

In the proposed work automatic humour recognition techniques have been applied to a conversational agent.

Humoristic features such as alliteration, antinomy and adult slang, are frequently present in humoristic sentences exchanged in real human conversations.

The recognition of such features allows a chat-bot to understand the humoristic context in a conversation and to properly react. The recognition techniques can be further on improved: a deeper analysis of texts, with the aim of searching other features, could improve the humour recognition.

Future work will regard the development of humoristic chat-bot speaking in different languages, exploiting other lexical resources such as the MultiWordNet project[16].

## References

[1] Dirk Heylen. Talking Head Says “Cheese!” Humor as an impetus for Embodied Conversational Agent Research CHI-2003 WorkShop: Humor Modeling In the Interface

[2] G. Ritchie. “Prospects for Computational Humour”. Pp. 283-291 in Proceedings of 7th IEEE International Workshop on Robot and Human Communication (ROMAN-98), Takamatsu, Japan, October 1998

[3] Attardo, S. Linguistic Theory of Humor. Mouton de Gruyter, Berlin, 1994

[4] R. Mihalcea and C.Strapparava. “Lerarning to laugh (automatically): Computational Models for Humor Recognition”. Computer Intelligence, Volume 22, 2006

[5] Alice Chabot: www.alicebot.org

[6] O. Stock and C.Strapparava. “Getting serious about the development of computational humor”. In proceedings of the 8th International Joint Conference on Artificial Intelligence (IJCAI-03) pp. 59-64, Acapulco, Mexico,2003

[7] J. McKay. “Generation of idiom-based witticisms to aid second language learning”. In Stock et al. (2002),pp. 77–87

[8] J. M. Taylor and L. J. Mazlack. “Computationally recognizing wordplay in jokes”. In Proceedings of Cognitive Science Conference, pages 2166–2171, Stresa, Italy (2004)

[9] G. Lessard and M. Levison. “Computational modeling of riddle strategies”. In ALLC/ACH Joint Annual Conference, Georgetown University, Washington, DC, pages 120–122, (1993)

[10] Julia M. Taylor, Lawrence J. Mazlack, "Toward Computational Recognition of Humorous Intent," Cognitive Science Conference 2005 Proceedings (CogSci 2005), Stresa, Italy, July, 2005, 2166-2171

[11] D. Buscaldi, P.Rosso. “Some Experiments in Humour Recognition Using the Italian Wikiquote Collection”, Applications of Fuzzy Sets Theory Lecture Notes in Computer Science, 2007

[12] W. Ruch. “Computer with a personality? Lessons to be learned from studies of the psychology of humor”. In Proceeding of The April Fools Day Workshop on Computational Humor, pp 57-70, University of Twente, the Netherlands,2002

[13] C. Bucaria . “Lexical and syntactic ambiguity as a source of Humor”. Humor 17(3):279-309, 2004

[14] G. Miller. “WordNet: A lexical database”. Communication of the ACM, 38(11):39-41, 1995

[15] CMU Dictionary: http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[16] MultiWordNet: http://multiwordnet.itc.it

[17] R. Mihalcea and C. Strapparava. “Bootstrapping for fun: Web-based construction of large data sets for humor recognition”. In Proceedings of the Workshop on Negotiation, Behaviour and Language (FINEXIN 2005), pp 25-30, Ottawa Canada