

# Pun Generation

## Vivaek Shivakumar

### Computer Systems Lab 2009-2010

## Abstract

Computational humor is a subfield of artificial intelligence focusing on computer recognition and generation of humorous texts. This paper investigates methods for generating various types of simple puns and jokes using a regular lexicon from the Natural Language Toolkit and the semantic web WordNet as well as phonetic information, which is the basis for most puns. Although no formal model or theory for humor or puns exists, pun-generation has been implemented using simple, constrained models and this paper attempts to recreate such implementations with possible expansions to more types of puns and even languages other than English.

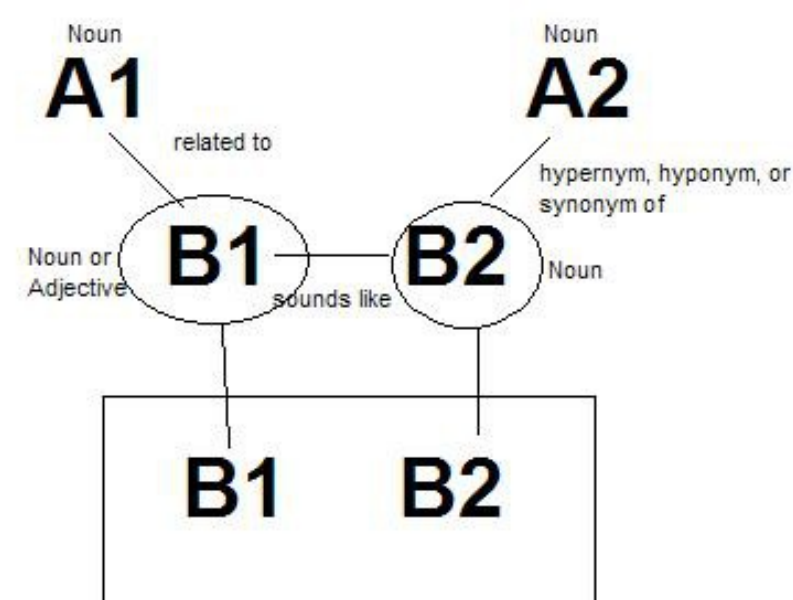


Fig 1: schema for a “cross” pun riddle

## Background and Introduction

Humor is a very relevant topic to artificial intelligence in the fields of natural language generation and computer creativity. The easiest type of humor for computer generation and recognition is the pun. Work has been done in pun generation and recognition since 1994 culminating in advanced user-interface programs such as STANDUP. The purpose of this project is to investigate and implement methods for generating various types of puns.

Humor has been studied extensively in a social or otherwise non-AI context, but no formal models or theories exist to be applied to AI. Nevertheless, the possibility of computer generation or recognition of humor is by no means impossible, as several computer programs have been created to generate small puns. These include JAPE, which was based on certain classes of puns modeled by schema and templates and utilizing the large lexicon WordNet, and STANDUP, a full interactive module for generating puns.

Finally, this project will try to apply the pun-generation techniques to languages other than English.

Examples of puns:

- What do you get when you cross a murderer with a breakfast food? A cereal killer.
- Pasteurize: too fast to see. (An example of redefinition wordplay)

Other types of puns and jokes include knock-knock jokes and Tom Swifty puns.

## Discussion

The current program uses WordNet from the Natural Language Toolkit and the CMU pronunciation dictionary of English to generate sets of words that fit the schema in Fig 1. Although the current schemata I have outlined are specific to a format of joke, they can potentially be generalized for greater efficiency of implementation when applying templates to create the actual pun riddle.

An example output from the program, with the input word (A1) “top”:

top-->peak --> peek-->peep

The question of the pun would involve “top” and “peep” and the answer would pun on the homophone nature of “peak” and “peek.”

## Results and Conclusions

Currently, some problems to resolve include:

- NLTK's inclusion of proper nouns, British variants, and obscure words
- The surface syntactic features to be adjusted, such as when to use articles
- Classifying nouns so that only certain types of nouns (e.g. living things, ideas, places) are used in certain contexts

Ideally, the project will be able to generate (with some success rate) various types of puns that are recognizably humorous, although not necessarily funny to the user. If success is found with English, the model can be applied to other language with semantic webs.