A PsychoLing Approach to Humor

David Kleinberg, Liam Creedon

May 3, 2019

Contents

1 Introduction

Humor has the power to break down social barriers, bridge cultural differences and, well, make people laugh. Seeing as it is something that even humans struggle with and what others are paid to perform professionally, can humor be modeled in such a way that computers can understand?

Comedic speech is a complex area to bring to the field of computational psycholinguistics, as it lends itself to deeply-rooted contextual language, involves a certain level of subjectivity to characterize something as *funny*, and often seems to lack a fixed set of guiding principles required to develop a model. The latter is perhaps the crux of this entire proposal and shortcomings of prior research done on the topic thus far. For instance, most - if not all - articles describing a model for humor narrowed the scope of all comedy to such a degree as to ruin the spontaneity of the subject in question itself. This broad flexibility of language is, by nature, what defines humor. Humor can take many forms - a joke, a story, or even a mere personality.

This project attempts to explore published computational models of humor to find a niche topic to contribute to this nascent field. More generally, it proposes comprehension and synthesis of comedy be included in the study of Computational Psycholinguistics as a whole, and provides a road map for doing so.

1.1 Motivations

Vinith Misra led a TED Talk [1] about the importance of developing the field of *Computational Humor*. Nowadays, computers exist in all aspects of our daily lives - from Watson, Siri, Cortana to even the most mundane of devices. Humor is the first step in a long series of evolutions for computers to relate to humans and vice versa. However, this effort is not only grounded in connecting with machines. It has far wider-spanning applications, which include deepening our understanding of personality and human interaction, and even has the power of connecting us with others. As for the latter, The Joking Computer[2] was developed as a program termed STANDUP [3] (**S**ystem **T**o **A**ugment **N**on-speaker's

Dialogue **U**sing **P**uns) to help underdeveloped children learn humor through play-on-words and emulating human interaction - something which might have been limited for these children. The original software is linked.

Other sources like Spike Jonze's movie *Her* [5] and Alex Garland's *Ex Machina* [6] illustrate the importance **personality** has in making artificial intelligence truly 'intelligent' and the conceivable effects/roles that human-like AI will play in a near-futuristic world - where humor is, inevitably, the first step of many in achieving something of this magnitude.

Humor is only one component of language that makes speech more 'human'.

1.2 Comprehension

One question that arises when studying comedy in the context of Cognitive Science - namely Computational Psycholinguistics is: can a computer be taught to laugh? Unpacking this question, touches on many layers such as the philosophical implications of human experience and includes, well, a layer of subjectivity as well. To some degree this is more subjective and less feasible than synthesizing humor - at least in terms of validation.

Computers lack tactile senses: The comedy that one might pay to watch such as stand-up involves the comic painting a picture to the audience which elicits some emotional or gut reaction and, of course, evokes an image. While a computer can attribute text, or even context, to an image (for example the Google Image Search), it cannot necessarily evoke an emotional & experiential response to that picture, which I would argue is important in understanding humor - it's quite a personal thing.

This begs the question: can language *alone* even encode humor? And of course, there are some cases where it does (i.e. puns, as seen in Types of Humor). But there are countless other instances where humor involves much *more* - perceiving sarcasm, for example, or meaning hidden behind the plain text.

A group from the University of Washington studied the computer's ability to identify humor in the form of Double Entendres [7] from the "That's What She Said" (TWSS) family of jokes popularized by the American Television series *The Office*.

Developing an algorithm to classify an utterance as either a factual statement or as alluding to a different context is complex and would require both deep semantic and cultural understanding to recognize the vast array of double entendres. This research team therefore narrows the context to only TWSS jokes. (DEviaNT = Double Entendre via Noun Transfer)

If a parse tree can be coined as a model's 'understanding', can grammars, lexicons, or contexts inform humor?

1.3 Synthesis

The advantages to the trajectory of synthesizing humor is the ease of verification. In some sense, humans can generally agree to something as a joke - even if they themselves don't necessarily find it funny. It also allows for incremental results, starting at a framework like **puns**, for example, and moving to something more contextually rooted and subtle.

Vinith goes on to describe a computational model (that one could term "algorithmic humor") where the base question is something to the effect of,

"what do you call a ____ ?"

where the "____" blank is a two word combination. And the response is also some two-word combination that the computer returns of similar meaning but with a comical twist. Of course, the majority of responses are terrible, but they could weed out some of comedic value. This is a scenario where computers have already made strides in generating novel humorous content.

A more sophisticated type of humor generating model was developed by Andrej Karpathy, who created Char RNN [4] (Github linked) to train a computer to imitate a dataset of choice and then generate new utterances from said source (e.g. Trump or Justin Bieber Tweets). Vinith describes this as painting a computer's personality with *data*. The humor exists in the *style*, not necessarily in a joke anymore.

1.4 Types of Humor

Determining the types of humor that exist is an important step in both understanding and synthesizing comedic speech. According to Business Insider, in an article based on the work of psychologist Rod Martin, there are four types of humor, which include: "affiliative, self-enhancing, aggressive, and self-defeating humor". These are further broken down in the Journal of Research in Personality [15].

However, one can conceivably think of many more types of humor than those discussed in the paper to the likes of: surprisal, racial, puns – these are perhaps the most widely studied (and, subjectively, the least humorous!), and **sarcasm** - just to name a few. The article goes on to share evidence that women and men vary in the ways they express humor. Further supporting the claim of subjectivity and lack of structure (mentioned earlier).

Understanding the different facets of humor is an important step in modeling humor computationally.

2 Proposal

The most viable approach to studying and computationally modeling humor, based on the research described throughout this proposal and the background developed in JHU's AS.050.360, an introductory course to the field of computational psycholinguistics, is to narrow the field of comedy to an extremely specific subset of jokes/humor so that the 'world' can be defined & modeled.

This project seeks to combine studies from the literature of TWSS (That's What She Said) jokes and "What do you call a... ?" synthesis model developed at the University of Aberdeen (described earlier). We propose further exploration of the models such as DEviaNT[7] to achieve a computer *generated* double entendre.

2.1 Approach

Our proposed model works as a basic output generator of a "That's what she said" (TWSS) joke. Our model would use two corpora to gather the primary sets of data.

The first data we are looking to collect is words (nouns, adjectives and verbs) that are more likely to appear in an entendre. That is, these words when used in accordance with our model, would produce a TWSS joke. For example, a word like "meat" is more likely to appear in a TWSS joke than a word like "folder".

To acquire this data, our model would be trained on two corpora, one that is composed of sentences that arise in common literature, and another that is formed from sentences that appear in erotic literature. For each word, we want to generate a likelihood that the word might appear in an erotic context versus. For nouns, we can base this on frequency of the overlap of this noun in both the standard and erotic corpora. We can also acquire similarity of nouns as related to the more explicitly sexual nouns. To do this, we would look at neighboring adjectives of the word, and match to nouns in the erotica corpus that also frequently appear next to those adjectives. For adjectives, we can take the examples, attached to nouns, that appear most frequently in just the erotic corpus. And for verbs, we can use words that are more likely to appear in erotic content, than non-erotic content.

The second data we are looking to collect is which sentence structures will allow our double entendres to share similar structure as utterances that might arise erotically. Basic structures are acquired by passing each sentence from the erotic corpus over a filter which looks for certain features. One approach would be based on the appearance of nouns, as defined from our data sets above. If the sentence contains any nouns that overlap with the set defined above, then we can raise the likelihood that this sentence's structure can be added to our data. Similarly, looking at the structural elements of a sentence, such as which adjectives and verbs appear, and what they are modifying can yield useful information, such as what actions are viable for our jokes.

Using the proper sentence structures acquired from above, as well as the nouns, adjectives and verbs that are most likely to appear in a standard literature, that insinuate sexual content, the model would be adeptly situated to generate a TWSS joke. Our model would use a randomly selected sentence structure, and then a randomly generated aspect of the sentence (i.e. the verb phrase "was licking", or the noun "snake"), and properly fill in the other blanks based on semantic similarity to the other parts of the sentence, as well as making sure that it is a logical, not directly sexual, sentence.

Now, with any model there will be errors and not every output would necessarily be a proper joke by our standards. The most sensible training that we could imagine would be to have humans sit down and determine which jokes were "funny" or at least logical in the confines of what a TWSS joke is. Some examples might come off as too explicitly sexual, which would not fit our confines.

3 Conclusion

Creating a computer model to generate double entendres - while quite seemingly insignificant in the scope of humor and personality - offers a strong basis for exploring the intricacies of language. Humans have the ability to account for context and infer subtext from very little. Implementing a model such as that described in the proposal) is a step towards reaching natural language synthesis and computer creativity.

There is no question as to whether the future will yield a computer or algorithm that can spit out something funny. Whether or not it will *understand* what makes something funny or *how* that happens (psychologically) is a question less likely to be answered in the near future.

References

[1] TED Talk on *Computational Humor*. Vinith Misra.
`https://ideas.ted.com/can-you-teach-a-computer-to-be-funny/`

[2] *The Joking Computer*. University of Aberdeen.
`http://joking.abdn.ac.uk/`

[3] The STANDUP Software Program. University of Aberdeen.
`https://www.abdn.ac.uk/ncs/departments/computing-science/software-480.php`

[4] Andrej Karpathy, *Char RNN* (recurrent neural network). Stanford University.
`https://github.com/karpathy/char-rnn`

[5] *Her* The Movie. Spike Jonze.
`https://www.imdb.com/title/tt1798709/`

[6] *Ex Machina* The Movie. Alex Garland.
`https://www.imdb.com/title/tt0470752/`

[7] *Deviance* The Double Entendre Software.
`https://www.aclweb.org/anthology/P11-2016`

[8] *Computational Humor*. Kim Binsted. University of Hawaii.
`https://pdfs.semanticscholar.org/ca45/535ad4e677fc9f1744e844d46a5e2ccb1cc8.pdf`

[9] G. Ritchie et al. 2007.
http://computationalcreativity.net/ijwcc07/papers/ritchie-et-al.pdf

[10] *Reverse Engineering Satire*. R West, E. Horvitz.
https://arxiv.org/abs/1901.03253

[11] *Joke Telling Robots are the Final Frontier of Artificial Intelligence.* Motherboard.
https://motherboard.vice.com/en_us/article/z43nke/joke-telling-robots-are-the-final-front

[12] Kao, Levy, & Goodman 2013.
http://www.mit.edu/ rplevy/papers/kao-levy-goodman-2013-cogsci.pdf

[13] Kao, Levy, & Goodman 2015.
http://www.mit.edu/ rplevy/papers/kao-etal-2015-cogsci-puns.pdf

[14] Spinter News. *It's hard work being funny - especially for robots.*
https://splinternews.com/its-hard-work-being-funny-especially-for-robots-1793853950

[15] Journal of Research in Personality.
https://www.sciencedirect.com/science/article/abs/pii/S0092656602005342