On Algorithmic Discovery and Computational Implementation of the Opposing Scripts Forming a Joke

Victor Raskin(✉)

Linguistics and CERIAS, Purdue University, West Lafayette, IN, USA
vraskin@purdue.edu

Abstract. The paper deals with the notion of 'script'. Scripts have been essential for the dominant formal theories of verbal humor since their inception in the late 1970s, and the formal theories gave rise to meaningful computational humor a decade or so later. Recent developments in computational semantics and computational humor have required a tighter definition of 'script' as a computational entity.

Keywords: Humor · Formal humor theory · Computational humor · Script

1 Introduction

The paper deals with the notion of 'script' and its readiness for computation, mostly in computational humor but also beyond. Computational humor does, of course, require the same computation as other meaning-based natural language processing applications (MB-NLP). Scripts formed the basis of the first full-fledged theory of humor, my Script-Based Semantic Theory of humor (SSTH: [1, 2]). They were kept on, without much elaboration, in Attardo's and my General Theory of Verbal Humor (GTVH: [3]). And it was not until the state of the art in computational semantics caught up with what the formal theory of humor needed for computation that the need to tighten up and to formalize, algorithmize, and compute actually arose, which led to the Ontological Semantic Theory of Humor (OSTH: [4]). In this section, we will introduce the notion of script first, as it was used in SSTH, which will be the very next item to be introduced, followed by GTVH and, finally, OSTH. We will then review computational humor, both how it should be done and why, and how it should not. The following sections will deal with the essence of the paper, i.e., how to algorithmize and compute scripts.

1.1 Scripts in Humor Theories

By the late 1970s, the notions of 'frame' and 'script' had been introduced at least in computer science [5, 6]. Neither had a formal definition, and its understanding depended on the reader's common sense: you know that a room has walls, a floor, a ceiling, always a door, often windows—so all of that is part of the frame for room. And

N. Streitz and P. Markopoulos (Eds.): DAPI 2015, LNCS 9189, pp. 671–679, 2015.
DOI: 10.1007/978-3-319-20804-6_61

the script of going to a restaurant includes these things that happen there: you are seated, offered the menus, etc. I did not feel compelled or enabled to go any further than that, so I referred to the script of DOCTOR and LOVER to explain how a most ordinary joke (1) worked:

(1) "Is the *doctor* at home?" the *patient* asked in his *bronchial* ***whisper***. "No," the *doctor*'s **young and pretty wife whispered** in reply. "**Come right in**."

The SSTH Main Hypothesis was that, to be a verbal joke, the text had to be compatible, in full or in part, with two opposing scripts. In the text above, the material that is compatible with the doctor script is italicized and the material compatible with the lover script boldfaced. The reader/hearer is strongly prompted towards the first script and pretty much ignores the material from the second script until he/she is stumped by the last-sentence punchline, which defeats, without any explanation like, *He will be back soon*, the patient's goal to get help. (Peter Derks' 1991 demo of the first MRI of his brain processing a joke actually demonstrated a total momentary collapse of all activity when the first script collapses and quickly recovers to handle the second script.) Then, the reader/hearer of (1) also quickly notices the second-script material and, given the premises and prejudices of the 1930s rural America, "gets" the joke.

This is the essence of SSTH, and one must understand the theoretical innocence of humor research at the time of its inception to figure out why the theory immediately gained the prominence it did. The theory did establish itself conditionally: it was supposed to work with a fully-fledged formal procedure of semantic representation. The whole pathos of the effort was that we can establish the joke potential of a text in the process of purely linguistic semantic processing. Only the semanticists, at least some of them, could appreciate the fact that such a processing was not then available, and there were no other semanticists among humor researchers.

Another reason for the immediate acceptance of the theory was that the field of humor research had been familiar with a purely household notion of theory, as in *I have a theory why Nicole left Jason*. Household wisdom rarely encompasses the philosophy of science, and the latter hardly ever discusses the properties of theories that are not of physics. The fact that theories must have purviews, premises, bodies, etc., that they must be falsifiable, justified, and evaluated, that they should have no unlisted exceptions [7] was amystery to humor researchers then and have remained a mystery to many since—see, for instance, the "theory" of benign violations [8], which is a typical Nicole/Jason partial observation about undefined entities. Much more seriously, see [9], which is essentially the same, except for coming from a major humor scholar who does know all there is to know about humor but not about real theories. So, the reason for SSTH's easily achieved prominence was that it was a real theory even though it was not fully described as such until several decades later.

The next phase of the linguistic theory of humor was the General Theory of Verbal Humor (GTVH: [3]), which came up with 6 Knowledge Resources (KRs), three of which were introduced to make a linguistic theory interdisciplinary: Situation (action theory), Target (sociology), Narrative Strategy (narratology)—see also [11]. Two KRs, Script Opposition and Language, encapsulated SSTH, virtually unchanged. And the last KR, Logical Mechanisms, the only one, whose status in the sequence-of-funnels hierarchy was not confirmed in a famous massive psychological experiment [10], remains mysterious.

The linguistic component of GTVH continued to affirm that linguistic semantics could account for the joke potential of a text except for script opposition, treated in both theories rather dismissively: [2] came up with a couple of very short lists of script oppositions, such as sex vs. no sex, life vs. death, money vs. no money, that covered a huge majority of all jokes; [12] made a somewhat more careful attempt to inventory script oppositions. Unlike its two predecessors, OSTH, which emerged a couple of decades later, could rely on a mature system of linguistic semantic representation and computation, and as such, it fully incorporated script opposition into the Ontological Semantic Technology [13–16]: in other words, OSTH, not yet fully deployed but following pretty much the same main hypothesis of script opposition, is supposed to be a working theory, and the way it is supposed to work constitutes the idea of real computational humor.

1.2 Computational Humor

Computational humor encompasses approaches and systems that enable the computer to detect and/or generate verbal humor. Real computational humor is based on computer understanding rather than on the mechanical use of word lists early on and machine learning later. The term 'real' is not meant evaluatively but only to signify that this is how humans use humor—with understanding. And the major consequence of computability of anything is the conclusion that humans have come up with a rigorous formal theory of the computed phenomenon, a full-fledged theory of the kind described in the previous section, that was good enough to be encoded in machine language and to enable the computer to detect nad/or generate humor. A theory like that probides revealing insights into humor, on the one hand, and becomes a component of true, big-issue artificial intelligence (AI).

2 Scripts

After Schanck cavalry assault on the restaurant script, compromised mostly by his denial of any role to syntax and insistence on reducing all actions to 11 primitives, there has been surprisingly little work on computing scripts. But then, of course, there has been very little work on computational semantics at all, in an era, almost completely monopolized by meaning-free machine learning (ML), whose algorithms self-perpetuated and metamorphosed into a powerful industry. Since around 2005, however, voices from inside the ML have arisen that berate its low precision [24], and consumers have been unhappy with ML-based NLP applications, especially the more ambitious and sophisticated applications like e-discovery in litigation. Some ML leaders have actually led the charge towards semanticalizing the industry at least somewhat, by adding elements of meaning and/or ontology [25, 26]. Scripts are likely to re-emerge soon as well, and they are already raising their messy heads in our own Ontological Semantic Technology.

2.1 Scripts in Ontological Semantic Technology?

Don't let us beat about the bush: there have been no scripts in Ontological Semantic Technology (OST) so far, as there were none in Ontological Semantics [27] *per se*, of which OST has been a much improved and more (and better) implemented revision—see Fig. 1.

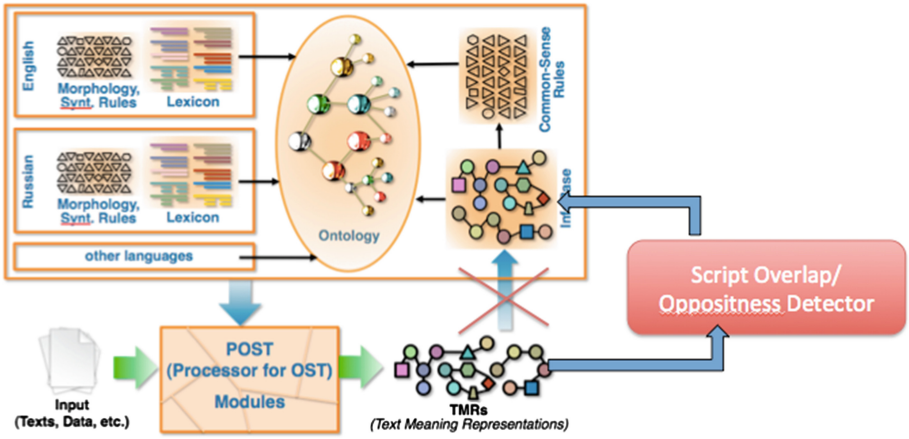

Fig. 1. OST architecture for OSTH

Let us ignore the OSTH part in the bottom left corner for the moment and focus on what OST is and what it does. The oval language-independent ontology in the center is indeed the main basis of the approach. It is an initially manually crafted hierarchy of concepts organized on the subsumption IS-A basis but linked with several hundred other properties. It is an engineering ontology [28], built mostly semi-automatically, with the help of a well-evolved and tested acquisition tool [14], without any philosophical claim that it reflects how the world is or the psychobiological claim that our brain contains the same ontology. Its only justification is that it works—in meaning representation and communication, especially in practical applications.

The main function of the ontology is to provide conceptual support for the lexical items in a language-specific lexicon: English and Russian are listed on the left as just examples of languages, as confirmed by the lower "other languages" block; they do happen to have been languages OST has been implemented on, joined by Spanish, Korean, Arabic, and Hebrew. Thus, the English word *drive*, in one of its verb senses, may be anchored in the concept GO whose INSTRUMENT is restricted to a concept like AUTOMOBILE or VEHICLE. And so will the corresponding sense of the Russian word *vesti*.

Also in the large Resource block, are the language-specific lexicons, where every sense of every lexical item—word, acronym, phrasal—is defined in terms of the anchoring concept and restricted properties, as shown above. Also there, there is the language-specific ecological, morphological, and syntactic information [27]. InfoBase is where all the successfully processed sentences go to make up the

language/knowledge experience of the system. And the common-sense rules, collected in processing.

The actual processing happens below the block: a text comes in, an OST software processes it in a variety of ways, and what comes out is a text-meaning representation (TMR), an ontological presentation of the meaning of the sentence. OST has used two different types of TMR calculation, pattern matching and graphic distances. The former one will present the English sentence Mary drove to Boston from New York yesterday, roughly, as in Fig. 2.

go	
agent	Mary
instrument	automobile
direction-from	New York
direction-to	Boston
time	yesterday

Fig. 2. Simplified OST TMR

The other software produces a graph TMR on the basis of the shortest distance between concepts in which the correct senses of the words in the sentence are anchored, and those graphs are much better visible to the computer than to human readers.

The successfully processed TMRs go to InfoStore. When the TMR fails the human engineer initiates the blame-assignment procedure, and that often diagnoses a common-sense failure, in which case a common-sense rule may be added to the Resource block. The rule may state, for instance, that one gets dressed before leaving home, and the rule, thus, supports an inference that people are dressed outside of their homes. The common-Common-Sense Rules block is, thus, the only place in OST where a whole sentence may—and does—typically appear. This is probably where scripts belong because they are also of that nature: if not common-sense rules, they are also part of our knowledge of the world.

2.2 Scripts as Language Entities

It was rather amazing to realize in the 1970s, when frames and scripts [5, 6] were introduced that it had not happened much earlier because human users definitely manipulate them all the time. As Schank could rely on his readers' knowledge of the script for attending a mid-toupscale restaurant, people fully dispose of a large number of ordinary scripts as well as developing scripts for shared experience, personally or professionally. Many scripts are culture-specific: thus the US morning routine differs from the continental European morning routine in a number of ways perhaps but most notably in the huge distinction between an American and continental breakfast. Couples develop sex routines. Colleagues establish meeting routines. There are fended bender routines, shopping routines, bill-paying routines, and so and so forth.

The most obvious way of handling a script is to present it as a set of sentences, each describing an individual attempt that is part of a script. This is what I must have felt intuitively when I invented a semi-formal presentation for the doctor and lover scripts when analyzing (1) in [1, 2]: a doctor was an adult human, who spent a considerable time at a medical school in the past and now sees patients, diagnoses them, and prescribes medication. A lover was an adult person, who has had sex at least once to a person of the (then) opposite sex, to whom he or she was not married. A bit more formally, something like the sequence of events in Figs. 3 and 4 must take place to establish X as a doctor and Y as a lover.

Figure 5 shows an abortive attempt to incorporate scripts into pre-OST Ontological Semantics [29] that OST has not yet picked up and incorporated. The if/then, and, and or logical operators had not, however, been actually incorporated into the system, even though [27] semi-tacitly allowed for them.

The scripts were developed for use in an application that would crawl the web and inform the officers of a company about the state of financial health of their partner companies, both suppliers and buyers. To my knowledge, such an application has not yet been implemented, and an expensive horde of human analysts provides an imperfect service. Obviously, an Ontological Semantic implementation would process the phrases and sentences into TMRs and develop a TMR-manipulating calculus for using scripts for inferencing and, more broadly, for reasoning.

2.3 Script Operations

Developing a TMR-manipulating calculus technically is a trivial algorithmic and programming task. Yet, inferencing and reasoning in NL, rather than in first-order logic as description logic does [30], is not simple, and the difficult part, as far as scripts are concerned, is script operations. What are they?

Obviously, it is a question whose significance goes far beyond computational humor but, almost equally obviously, for people in computational humor, this field can help to establish some helpful prompts for any form of HCI. One obvious exploitation of scripts in humor is the ability to mention scripts and to pretend to establish scripts, as in Jokes (2–3), respectively.

(2) It was such a hurried morning for me that I almost burned the truffle lasagna for the kids.

(3) Two Russian peasants chat over the fence between their outhouses early in the morning. “Ouch,” one of them says, “The sun is almost up, and my cow has not been milked yet.” “Nor has my woman been fucked yet,” adds the other.

Similarly, in the financial world, a casual remark that a company applied for an unusual loan will bring up the much-feared specter of bankruptcy. But, then again, this may turn out to be the wrong conclusion. In spite of this and other difficulties with script operations, the initial problem is script acquisition. In [27], the complex event of teach is analyzed at length, setting up various sub-events. My co-author insisted on avoiding the notion of script there, and the legitimate part of his reservation was the finer grain size of that script. I had panicked when my over-enthusiastic Ph.D. students on the soft side started talking about the scripts of life or of poetry—that was much too

X is a doctor if and only if:

1. X went to an accredited medical school and graduated from it.
2. X passed an extended internship
3. X was licensed as a physician
4. X has opened or joined a medical practice or a hospital
5. X treats patients on a regular basis by examining or listening to them, diagnosing their condition and sending them to tests or specialists and/or prescribing them medication

Fig. 3. "Script" for doctor

Y is a lover if and only if:

1. Y is a teenager or older
2. There is a Z of the opposite sex who is a teenager or older
3. Y and Z are not married to each other
4. Y and Z have had sex at least once

Fig. 4. "Script" for lover.

APPROACH-BANKRUPTCY

If Or company has cash problems
company can't meet payroll
company misses loan payment
company seeks loan
Then company maynear bankruptcy

DECLARE-BANKRUPTCY

If company declares bankruptcy
And company files for Chapter 11
Or court appointsreceiver for company
Then And company officers lose control
company operates under receiver
Orcompany stops operating
company liquidates assets
creditors get partial payment

Fig. 5. Two bankruptcy scripts

course. The fact that just about any event can be analyzed into sub-events arouses the fear of infinite regress. Not only does the joke punch line, largely not counterparted in non-humorous text, help to focus on the main script opposition in a joke [31] but it also establishes the appropriate grain size of the scripts.

References

1. Raskin, V.: Semantic mechanisms of humor. In: Chiarello, C., et al. (eds.) Proceedings of the Fifth Annual Meeting, Berkeley Linguistics Society. University of California, Berkeley (1979)
2. Raskin, V.: Semantic mechanisms of humor. D. Reidel, Dordrecht, Boston, Lancaster (1985)
3. Attardo, S., Raskin, V.: Script theory revis(it)ed: joke similarity and joke representation model. Humor Int. J. Humor Res. **4**, 293–348 (1991). doi:10.1515/humr.1991.4.3-4.293
4. Raskin, V., Hempelman, C., Taylor, J.: How to understand and assess a theory: the evolution of the SSTH into the GTVH and now into the OSTH. J. Literary Theor. **3**, 285–312 (2009)
5. Minsky, M.: A framework for representing knowledge. In: Winston, P.H. (ed.) The Psychology of Computer Vision, pp. 211–277. McGrow Hill, New York (1975)
6. Schank, R., Abelson, H.: Scripts, Plans, Goals and Understanding. Erlbaum, Hillsdale (1977)
7. Raskin, V.: A little metatheory: thoughts on what a theory of computational humor should look like. In: AAAI Symposium on the Artificial Intelligence of Humor, Arlington, VA (2012)
8. McGraw, P., Warren, C.: Benign violations: making immoral behavior funny. Psychol. Sci. **21**, 1141–1149 (2010)
9. Oring, E.: Parsing the joke: the general theory of verbal humor and appropriate incongruity. Humor Int. J. Verbal Humor **24**(2), 203–222 (2011)
10. Ruch, W., Attardo, S., Raskin, V.: Towards an empirical verification of the general theory of verbal humor. Humor Int. J. Humor Res. **6**(2), 123–136 (1993)
11. Taylor, J., Raskin, V.: Towards the Cognitive Informatics of Natural Language: The Case of Computational Humor. Int. J. Cogn. Inform. Nat. Intell. **7**, 3 (2013)
12. Raskin, V.: Linguistic heuristics of humor: a script-based semantic approach. In: Apte, M. (ed.) Language and Humor, Int. J. Sociol. Lang. (Special Issue) **65**(3), 11–26 (1987)
13. Raskin, V., Hempelmann, C., Taylor, J.: Guessing vs. knowing: the two approaches to semantics in natural language processing. In: Annual International Conference Dialogue (2010)
14. Taylor, J., Hempelmann, C., Raskin, V.: On an automatic acquisition toolbox for ontologies and lexicons in ontological semantics. In: International Conference in Artificial Intelligence: ICAI 2010 (2010)
15. Hempelman, C., Taylor, J., Raskin, V.: Application-guided ontological engineering. In: International Conference in Artificial Intelligence: ICAI 2010 (2010)
16. Taylor, J., Raskin, V., Hempelmann, C.: From disambiguation failures to common-sense knowledge acquisition: one day in the life of an ontological semantic system. In: Web Intelligence Conference (2011)
17. Raskin, V., Attardo, S.: Non-literalness and non-bona-fide in language: approaches to formal and computational treatments of language. Cogn. Pragamtics **2**, 1 (1994)
18. Binstead, K., Ritchie, G.: An implemented model of punning riddles. In: 12th National Conference on Artificial Intelligence (1994)
19. Stock, O.: Password swordfish: verbal humor in the interface. In: Hulstijn, J., Nijholt, A. (eds.) Proceedings of the International Workshop on Computational Humor, TWLT 12. University of Twente, Enschede (1996)
20. Wilks, Y.: Artificial companions. In: Bengio, S., Bourlard, H. (eds.) MLMI 2004. LNCS, vol. 3361, pp. 36–45. Springer, Heidelberg (2005)
21. Rosenthall, S., McKeown, K.: Sentiment detection of subjective phrases in social media. http://www.aclweb.org/anthology/S/S13/S13-2079.pdf

22. Mihalcea, R., Strapparava, C.: Computational laughing: automatic recognition of humorous one-liners. In: Cognitive Science Conference, pp. 1533–1538 (2005)
23. Mihalcea, R., Strapparava, C.: Learning to laugh (automatically): computational model for human recognition. Comput. Intell. **22**(2), 126–142 (2006)
24. Schiffman, B., McKeown, K.: Context and learning in novelty detection. In: HLT/EMNLP (2005)
25. Moldovan, D., Badulescu, A., Tatu, M., Antohe, D., Girju, R.: Models for the semantic classification of noun phrases. In: HLT/NAACL Workshop on Computational Lexical Semantics, pp. 60–67 (2005)
26. Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Betteridge, J., Carlson, A., Dalvi, B., Gardner, M., Kisiel, B., Krishnamurthy, J., Lao, N., Mazaitis, K., Mohamed, T., Nakashole, N., Platanios, E., Ritter, A., Samadi, M., Settles, B., Wang, R., Wijaya, D., Gupta, A., Chen, X., Saparov, A., Greaves, M., Welling, J.: Never ending learning. In: Conference on Artificial Intelligence, AAAI 2015 (2015)
27. Nirenburg, S., Raskin, V.: Ontological Semantics. MIT Press, Cambridge (2004)
28. Gruber, T.: Toward principles for the design of ontologies used for knowledge sharing. In: Guarino, N., Poli, R. (eds.) The Role of Formal Ontology in the Information Technology, Int. J. Hum. Comput. Stud. (Special Issue) **43**(5–6), 907–928 (1995)
29. Raskin, V., Nirenburg, S., Hempelmann, C., Nirenburg, I., Triezenberg, K.: The genesis of a script for bankruptcy in ontological semantics. In: Hirst, G., Nirenburg, S. (eds.) Proceedings of the Text Meaning Workshop, HLT/NAACL 2003: Human Language Technology and North American Chapter of the Association of Computational Linguistics Conference. ACL, Edmonton (2003)
30. Baader, F., Calvanese, D., McGuiness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook, pp. 47–100. Cambridge University Press, Cambridge (2003)
31. Taylor, J.: Different knowledge, same joke: response-based computational detection of humor. In: Streitz N., Markopoulos, P. (eds.) DAPI 2015. LNCS, vol. 9189, pp. xx–yy. Springer, Heidelberg (2015)