Creative Introspection and Knowledge Acquisition:
Learning about the World Through
Introspective Questions and Exploratory Metaphors

Tony Veale, Guofu Li

School of Computer Science and Informatics, University College Dublin, Belfield, Dublin D4, Ireland.

{tony.veale, li.guofu.l}@gmail.com

Abstract

Introspection is a question-led process in which one builds on what one already knows to explore what is possible and plausible. In creative introspection, whether in art or in science, framing the right question is as important as finding the right answer. Presupposition-laden questions are themselves a source of knowledge, and in this paper we show how widely-held beliefs about the world can be dynamically acquired by harvesting such questions from the Web. We show how metaphorical reasoning can be modeled as an introspective process, one that builds on questions harvested from the Web to pose further speculative questions and queries. Metaphor is much more than a knowledge-hungry rhetorical device: it is a conceptual lever that allows a system to extend its model of the world.

1. Introduction

Picasso once remarked that "*Computers are useless. They can only give you answers*". Though reflecting a blinkered view of computers, his aphorism skewers a widespread tendency to prize the best answers while taking the best questions for granted. Creative processes, in art and in science, are fundamentally introspective (see Boden, 1994), and to find the right answers one must learn to ask the right questions. Indeed, because questions often presuppose a shared understanding of the world, these presuppositions are a rich source of knowledge even when the questions go unanswered. We show here how knowledge can be acquired from the Web by harvesting presupposition-laden questions, and show how these questions can in turn be used as the basis for further introspection via metaphor.

Questions are the basic currency of any creative system. Consider the introspective workings of a metaphor processing system. To generate metaphors for a topic T, a system must identify candidate vehicles V by asking "What Vs are most like T?", and for each V, "how is T like V?".

Moreover, a system must appreciate the most salient aspects of V, since any metaphor *T is V* will transfer the most stereotypical aspects of V to T (see Ortony, 1979). We can identify these aspects by looking for questions that are commonly asked about V, and at their presuppositions (e.g. "*why do pirates wear eye patches*" presupposes that pirates wear eye patches). For purposes of introspective reasoning, the questions asked about V can be considered a good representation of V, and so we expect two concepts V_1 and V_2 to be similar to the extent that the same questions are asked of both. Likewise, the metaphor *T is V* prompts to us to take the questions we normally ask of V and introspectively ask them of T. Those that can be answered affirmatively can then become part of our representation of T. Metaphorical introspection allows us to leverage what we know about a topic and transfer this knowledge to new domains. We show here how questions can be harvested from, and answered on, the Web, to encode old knowledge and elicit new knowledge. By trading in questions as a knowledge representation, a system can learn how to pose its own questions and extend this representation as needed.

We develop and test these ideas in the following sections. A brief survey of related work is presented in section 2, while section 3 describes the basic question-harvesting mechanism and the use of questions as a knowledge representation. Section 4 then shows how this question-based representation is leveraged in metaphor comprehension and generation. This section also shows how metaphor is used to flesh out an under-developed concept, and introduces the notion of a knowledge *mash-up*, a composite representation of a topic that is based on the introspective application of its most likely metaphors. An empirical evaluation of these ideas is presented in section 5. Here we examine the effectiveness of using questions as a knowledge-representation, of using the Web to resolve the many new questions that are posed during metaphor processing, and of using metaphor and mash-ups as an introspective means of knowledge acquisition. The paper concludes with some observations about future directions in section 6.

2. Related Work and Ideas

Metaphor and knowledge representation are tightly coupled phenomena. It takes knowledge to create or comprehend a metaphor, while metaphor allows us to bend and stretch our knowledge into new forms and niches. The computational treatment of metaphor thus presents a diverse catalogue of flexible representation schemes. Wilks (1978) argues that since most metaphors are semantically anomalous, a malleable *preference semantics* is required rather than a set of brittle semantic constraints. Fass (1991) builds on preference semantics to define a frame-based *collative semantics* that allows an alternative literal meaning to be salvaged from a figurative anomaly. Martin (1990) builds on the work of Lakoff and Johnson (1980) to show how conventional metaphors (like "*to catch a cold*" and "*kill a process*") can be modeled in an AI knowledge representation, and then extended as needed to interpret new variations on the same metaphors in a given domain (e.g. "*to kill Emacs*" in the Unix domain). Way (1991) argues that metaphor requires a *Dynamic Type Hierarchy* (DTH) that can dynamically create new categories as they are needed. A rigid taxonomy like that of WordNet (Fellbaum, 1998) may be useful for literal language, but is unsuited to the demands of metaphor. However, a WordNet-scale realization of Way's DTH remains an elusive goal.

We use metaphors not just as rhetorical flourishes, but as a basis for extending our inferential powers into new domains (Barnden, 2006). Indeed, work on analogical metaphors shows how metaphor and analogy use knowledge to create knowledge. Gentner's (1983) *Structure-Mapping Theory* (SMT) argues that analogies allow us to impose structure on a poorly-understood domain, by mapping knowledge from one that is better understood. SME, the *Structure-Mapping Engine* (Falkenhainer *et al.,* 1989), implements these ideas by identifying sub-graph isomorphisms between two mental representations. SME then projects connected sub-structures from the source to the target domain. SMT prizes analogies that are systematic, yet a key issue in any structural approach is how a computer can acquire structured representations for itself.

The availability of large corpora and the Web suggests a means of relieving the knowledge bottleneck that afflicts computational models of metaphor and analogy. Turney and Littman (2005) show how a statistical model of relational similarity can be constructed from Web texts for handling proportional analogies of the kind used in SAT and GRE tests. No hand-coded or explicit knowledge is employed, yet Turney and Littman's system achieves an average human grade on a set of 376 SAT analogies (such as *mercenary:soldier::?:?* where the best answer among four alternatives is *hack:reporter*). Almuhareb and Poesio (2004) describe how attributes and values can be harvested for word-concepts from the Web, showing that these properties allow word-concepts to be clustered into category structures that replicate the semantic divisions made by WordNet. Veale and Hao (2007a) describe how stereotypical knowledge can be acquired from the Web by harvesting similes of the form "as P as C" (as in "*as smooth as silk*"),

and go on to show, in Veale and Hao (2007b), how this body of 4000 or so stereotypes can be used in a Web-based model of metaphor generation and comprehension.

Shutova (2010) combines elements of several of these approaches. She annotates verbal metaphors in corpora (such as "to *stir* excitement", where the verb "stir" is used metaphorically) with the corresponding conceptual metaphors identified in Lakoff and Johnson (1980). Statistical clustering techniques are then used to generalize from the annotated exemplars, allowing the system to recognize other metaphors in the same vein (e.g. "he *swallowed* his anger"). These clusters can also be analyzed to identify literal paraphrases for a given metaphor (such as "to *provoke* excitement" or "*suppress* anger"). Shutova's approach is noteworthy for operating with Lakoff and Johnson's inventory of conceptual metaphors without actually using an explicit knowledge representation.

The questions people ask, and the Web queries they pose, are a rich source of implicit knowledge about the world. The challenge we face as computationalists lies in turning this implicit knowledge into explicit representations. Pasca and Van Durme (2007) show how knowledge of classes and their attributes can be extracted from the queries that are processed (and logged) by Web search engines. We focus here on well-formed questions, found either in the query logs of a search engine or harvested from the texts of the Web. These questions can be viewed as atomic properties of their topics, but can also be parsed to yield logical forms for reasoning. We show here how, by representing topics via the questions that are asked about them, we can also grow our KB via metaphor, by posing these questions introspectively of other topics as well.

3. Eavesdropping for Questions on the Web

Amid the ferment and noise of the Web sit nuggets of stereotypical world knowledge, in forms that can be automatically extracted by a computer. To acquire a property P for a topic T, one can look for explicit declarations of T's *P-ness*, but such declarations are rare, as speakers are loathe to explicitly articulate truths that are tacitly assumed by listeners. Hearst (1992) observes that the best way to capture tacit truths in large corpora (or on the Web) is to look for stable linguistic constructions that presuppose the desired knowledge. So rather than look for "*all Xs are Ys*", which is logically direct but exceedingly rare, *Hearst*-patterns like "*Xs and other Ys*" presuppose the same hypernymic relations. By mining presuppositions rather than declarations, a harvester can cut through the layers of noise and misdirection that are endemic to the Web.

If W is a count noun denoting a topic T_W, then the query "*why do W+plural *"* allows us to retrieve questions posed about T_W on the Web, in this case via the Google™ API.

If W is a mass noun or a proper-name, we instead use the query "*why does W *"*. These two formulations show the benefits of using questions as extraction patterns: a query is framed by a WH-question word and a question mark, ensuring that a complete statement is retrieved (Google

snippets often contain sentence fragments); and number agreement between "do"/"does" and W suggests that the question is syntactically well-formed (good grammar helps discriminate well-formed musings from random noise). Queries with the subject T_W are dispatched whenever the system wishes to learn about a topic T. We ask the Google API to return 200 snippets per query, which are then parsed to extract well-formed questions and their logical forms. Questions that cannot be so parsed are rejected as being too complex for later re-use in introspection.

For instance, the topic *pirate* yields the query "*why do pirates **", to retrieve snippets that include these questions:

Why do pirates always have parrots on their shoulder?
Why do pirates wear eye patches?
Why do pirates hijack vessels?
Why do pirates have wooden legs?

Parsing the 3rd question above, we obtain its logical form:

$$\forall x \; pirate(x) \rightarrow \exists y \; vessel(y) \wedge hijack(x, y)$$

Questions are retrieved on a need-to-know basis for a given topic. However, a system needs a critical mass of commonsense knowledge before it can be usefully applied to problems such as metaphor comprehension and generation, or to other similarity-centered tasks that presuppose a large body of knowledge that one can draw on for comparisons. Ideally, we could extract a large body of everyday musings from the query log of a search engine like Google, since many users persist in using full NL questions as Web queries. Yet such logs are jealously guarded, not least on concerns about privacy. Nonetheless, engines like Google do expose the most common queries in the form of text completions: as one types a query into the search box, Google anticipates the user's query by matching it against past queries, and offers a variety of popular completions.

In an approach we dub Google *milking*, we coax completions from the Google search box for a long list of strings with the prefix "why do", such as "why do a" (which prompts "*why do a*nimals hibernate?"), and "why do aa" (which prompts "*why do aa* batteries leak?"). We use a manual trie-driven approach, using the input "why do X" to determine if any completions are available for a topic prefixed with X, before then drilling deeper with "*why do X*a" … "*why do X*z". Though laborious, this process taps into a veritable mother lode of nuggets of conventional wisdom. Two weeks of milking yields approx. 25,000 of the most common questions on the Web, for over 2,000 topics, providing critical mass for the processes to come.

4. Introspective Metaphors and *Mash-ups*

A system that finds knowledge in questions posed by others is ideally poised to ask questions of its own, by repurposing past questions for new topics. Consider that Google milking yields the following common questions for *poet*:

Why do poets repeat words?
Why do poets use metaphors?

Why do poets use alliteration?
Why do poets use rhyme?
Why do poets use repetition?
Why do poets write poetry?
Why do poets write about love?

Questioning the Web directly, the system finds other common presuppositions about poets, such as "*why do poets die poor?*" and "*why do poets die young?*", precisely the kind of knowledge that shapes our stereotypical view of the topic yet which one is unlikely to find in a dictionary or other lexico-semantic resources. Now imagine we pose the metaphor *Philosophers are Poets*, which prompts the introspective question "*how are philosophers like poets?*". This question spawns others, which are produced by replacing the subject of the *poet*-specific questions above, yielding new introspective questions such as "*do philosophers write poetry?*", "*do philosophers use metaphors?*", and "*do philosophers write about love?*". Each repurposed question can be answered by again appealing to the Web: the system simply looks for evidence that the hypothesis in question (such as "*philosophers use metaphors*") is used in one or more Web texts. In this case, the Google API finds supporting documents for the following hypotheses: "*philosophers die poor*" (3 results), "*philosophers die young*" (6 results), "*philosophers use metaphors*" (156 results), and "*philosophers write about love*" (2 results). In line with Ortony (1979), the goal here is not to show that these behaviors are as salient for philosophers as they are for poets, rather that they are meaningful for philosophers.

4.1 Generative Similarity and Metaphor

In metaphor generation, one starts with a topic T and introspects about the vehicles V_1 … V_n that might plausibly yield a meaningful and revealing comparison. A locality assumption limits the scale of the search space for vehicles, by assuming that T must exhibit a pragmatic similarity to any vehicle V_i. Budanitsky and Hirst (2006) describe a raft of term-similarity measures based on WordNet (Fellbaum, 1998), but what is needed for metaphor is a generative measure: one that can quantify the similarity of T to V as well as suggest a range of likely V's for any given topic T.

We construct such a measure via corpus analysis, since a measure trained on corpora can easily be made corpus-specific and thus domain- or context-specific. The Google ngrams (Brants and Franz, 2006) provide a large collection of word sequences from Web texts. Looking to the 3-grams, we extract coordinations of generic nouns of the form "Xs and Ys". For each coordination, such as "*tables and chairs*" or "*artists and scientists*", X is considered a pragmatic neighbor of Y, and vice versa. When generating metaphors for a topic T, we now consider the pragmatic neighbors of T to be candidate vehicles for comparison.

Further, if we consider the pragmatic neighbors of T to

be features of T, then a vector space representation for topics can be constructed, such that the vector for a topic T contains all of the neighbors of T that are identified in the Google 3-grams. In turn, this vector representation allows us to calculate the similarity of a topic T to a vehicle V, and thus rank the neighbors of T by their similarity to T.

This approach to similarity does not use WordNet, but is capable of replicating the same semantic divisions made by WordNet. Recall that Almuhareb and Poesio (2004) extracted features for concepts from text-patterns found on the Web. These authors tested the efficacy of the extracted features by using them to cluster 214 words taken from 13 semantic categories in WordNet (henceforth this experimental setup is denoted AP214), and report an accuracy of **0.85** in replicating the category structures of WordNet. But if the pragmatic neighbors of a term are instead used as features for that term, and if a term is also considered to be its own neighbor, then an even higher accuracy of **0.934** is achieved on AP214. Indeed, using pragmatic neighbors as features in this way requires a vector space of just 8,300 features for AP214, whereas Almuhareb and Poesio's original approach to AP214 used approx. 60,000 features.

Intuitively, writers use the pattern "Xs and Ys" to denote an ad-hoc category, so topics linked by this pattern are not just similar but truly comparable, and perhaps interchangeable. Choices of vehicle for T are ranked by their perceived similarity to T, as described above. Thus, when generating metaphors for *philosopher*, the most highly ranked vehicles suggested via the Google 3-grams are: *scholar, epistemologist, ethicist, moralist, naturalist, scientist, doctor, pundit, savant, explorer, intellectual* and *lover*.

4.2 Mixed Metaphors and Conceptual Mash-ups

The problem of finding good metaphors for a topic T is highly under-constrained, and precisely which neighbor of T to use as a metaphorical vehicle for T will depend on the contextual goals of the speaker. However, when metaphor is used introspectively for knowledge acquisition, we can make use of a context-free structure dubbed a *conceptual mash-up*. If V_1 ... V_n are the n closest neighbors of T as ranked by similarity to T, then a mash-up can be constructed to describe the semantic potential of T by collating all of the questions from which the system derives its knowledge of V_1 ... V_n, and by repurposing each for T. A complete mashup collates questions from all the neighbors of a topic, while a 10-neighbor mashup for *philosopher*, say, would collate all the questions possessed for *scholar ... explorer* and then insert *philosopher* as the subject of each. In this way a conceptual picture of *philosopher* could be created, by drawing on beliefs such as *naturalists tend to be pessimistic* and *humanists care about morality*.

A 20-neighbor mashup for *philosopher* would also integrate the system's knowledge of *politician* into this picture, to suggest e.g. that *philosophers lie, philosophers cheat,*

philosophers equivocate and even that *philosophers have affairs* and *philosophers kiss babies*. Each of these hypotheses can be put to the test in the form of a Web query; thus, the hypotheses "*philosophers lie*" (586 Google hits), "*philosophers cheat*" (50 hits) and "*philosophers equivocate*" (11 hits) are each validated via Google, whereas "*philosophers kiss babies*" (0 hits) and "*philosophers have affairs*" (0 hits) are not. As one might expect, the most domain-general hypotheses show the greatest promise of taking root in a target domain. Thus, "*why do artists use Macs?*" is more likely to be successfully transferred in a metaphor than "*why do artists use perspective drawing?*".

The generality of a question is related to the number of times it appears in our knowledge-base with different subjects. Thus, "*why do ___ wear black*" appears 21 times, while "*why do ___ wear black hats*" and "*why do ___ wear white coats*" each just appear twice. When a mash-up for a topic T is presented to the user, each imported question Q is ranked according to two criteria: Q_{count}, the number of neighbors of T that suggest Q; and Q_{sim}, the similarity of T to its most similar neighbor that suggests Q. Both can be combined into a single salience measure $Q_{salience}$ as in (1):

$$(1) \qquad Q_{salience} \; = \; Q_{sim} * Q_{count} / (Q_{count} + 1)$$

It is time-consuming to test every question in a mash-up against Web content, as a mash-up of *m* questions requires *m* Web queries. It is more practical to choose a cut-off *w* and simply test the top *w* questions, as ranked by salience in (1). In the next section we evaluate the ranking of questions in a metaphor or mash-up, and estimate the likelihood of successful knowledge transfer from one topic to another.

5. Empirical Evaluation

The locality assumption constrains the number of vehicles that can contribute to a metaphor or mash-up. Knowledge of a vehicle V can be transferred to topic T only if V and T are pragmatic neighbors, as identified via corpus analysis. Yet, the Google 3-grams suggest a wealth of neighboring terms, so locality does not unduly hinder the transfer of knowledge. Consider a test-set of 10 common terms, *artist, scientist, terrorist, computer, gene, virus, spider, vampire, athlete* and *camera*, where knowledge harvested for each of these terms (see section 3) is introspectively transferred to all of their pragmatic neighbors. For instance, "*why do artists use Macs?*" suggests "*musicians use Macs*" as a hypothesis, which is validated by 5,700 Web hits. In total, 410,000 hypotheses are generated from these 10 terms, and when posed as Web queries to validate their content, approx. 90,000 (21%) are validated by usage in Web texts.

Knowledge tends to cluster into pragmatic neighborhoods (this, after all, is the basis of categorization), and hypotheses likewise tend to be validated in clusters. As illustrated in Figure 1, the probability that a hypothesis is valid for a given topic grows with the number of neighbors for which it is already believed to be valid (Q_{count}).
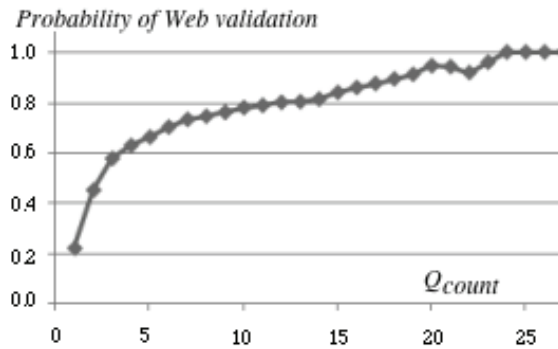
Figure 1. *Likelihood of a hypothesis in a metaphor or mash-up being validated via Web search (y-axis) for hypotheses suggested by Q_{count} neighbors (x-axis).*

Unsurprisingly, close pragmatic neighbors with a high similarity to the topic exert a greater influence than more remote neighbors. Figure 2 shows that the probability of a hypothesis for a topic being validated by Web usage grows with the number of the topic's neighbors that suggest it *and* its similarity to the closest of these neighbors ($Q_{salience}$).
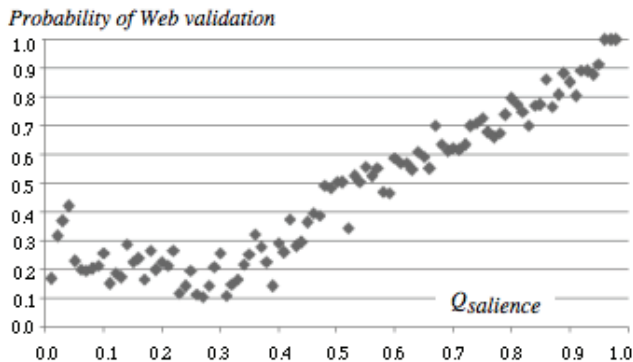

Figure 2. *Likelihood of a hypothesis in a metaphor or mash-up being validated via Web search (y-axis) for hypotheses with a particular $Q_{salience}$ measure (x-axis).*

In absolute terms, hypotheses perceived to have high salience (e.g. > .6) are much less frequent than those with lower ratings. So a more revealing test is the ability of the system to rank the hypotheses in a metaphor or mash-up so that the top-ranked hypotheses have the greatest likelihood of being validated on the Web. That is, to avoid information overload, the system should be able to distinguish the most plausible hypotheses from the least plausible, just as search engines like Google are judged on their ability to push the most relevant hits to the top of their rankings.

Figure 3 shows the average rate of validation for the top-n hypotheses (as ranked by perceived salience) of complete mash-ups generated for each of our 10 test terms from *all* of their neighbors. Since these are common terms, they have many neighbors that suggest many hypotheses. On average, about 85% of the top 20 hypotheses in each mash-up are validated on the Web as plausible, while just 1 in 4 of the top 60 hypotheses in a mashup is not Web-validated.
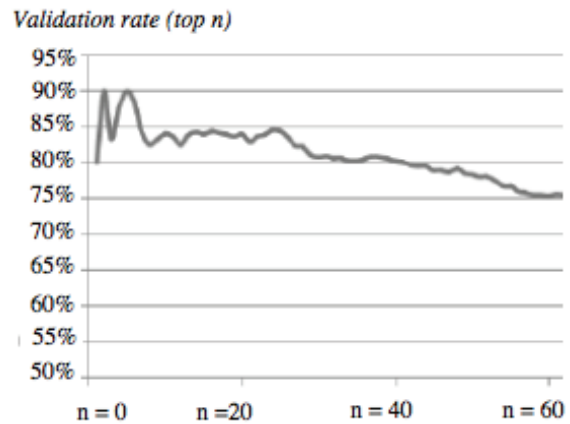

Figure 3. *Average % of top-n hypotheses in a mash-up (as ranked by $Q_{salience}$) that are validated by Web search.*

Figures 1 – 3 show that the system is capable of extracting knowledge from the Web (section 3) which can be successfully transferred to neighboring terms via metaphors and mashups (section 4), and then meaningfully ranked by salience. But just how useful is this knowledge? To determine if it is the kind of knowledge that is useful for categorization – and thus the kind that captures the perceived essence of a concept – we use it to replicate the AP214 categorization test of Poesio and Almuhareb (2004). Recall that AP214 tests the ability of a feature–set / representation to support the category distinctions imposed by WordNet, so that 214 words can be clustered back into the 13 WordNet categories from which they are taken. Thus, for each of these 214 words, we harvest questions from the Web, and treat each question body as an atomic feature of its subject.
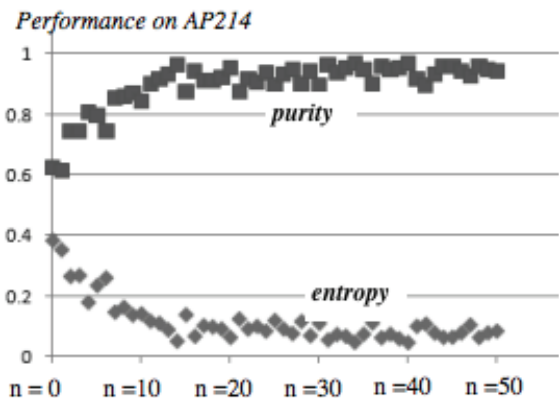

Figure 4. *Performance on AP214 improves as more knowledge is transferred from the **n** closest neighbors of a term.*

Clustering over these features alone offers poor accuracy when reconstructing WordNet categories, yielding a cluster purity of just over 0.5. One AP214 category in particular, for time units like *week* and *year*, offers no traction to the question-based approach, and accuracy / purity increases to 0.6 when this category is excluded. People, it seems, rarely

question the conceptual status of an abstract temporal unit.

But as knowledge is gradually transferred to the terms in AP214 from their pragmatic neighbors, so that each term is represented as a mash-up of its n nearest neighbors, categorization markedly improves. Figure 4 shows the increasing accuracy of the system on AP214 (excluding the vexing *time* category) when using mashups of increasing numbers of neighbors. Metaphors really do bolster our knowledge of a topic with insights that are relevant to categorization.

6. Concluding Remarks

Inside every "why do" or "how do" question sits a simpler "do" question which presupposes an affirmative answer. One needs world knowledge to pose such questions, but conveniently for computationalists, this knowledge is made obvious in the form of the question itself. So a computer can acquire this knowledge simply by eavesdropping on other people's questions, and in doing so, learn the questions that are most salient for a given topic. A system that acquires its knowledge from the questions of others also learns how to introspect, insofar as it learns how to pose meaningful questions of its own for similar / related topics.

We have shown here how questions can provide the world knowledge needed to drive a robust, empirically-founded model of metaphor processing. The ensuing powers of introspection, though basic, can be used to speculate upon the conceptual make-up of a given topic, not only in individual metaphors but in rich, informative mash-ups. The Web is central to this approach: not only are questions harvested from the Web, but newly introspected hypotheses are also validated by means of simple Web queries. The resulting approach is practical, robust and quantifiable, and uses an explicit knowledge representation that can be acquired on demand for a given topic. Most importantly, this approach makes a virtue of metaphor, and argues that computationalists should study metaphor not as a problem of language but as a *tool* of thought, one that can be used to leverage knowledge on a computer just as in the mind.

We have presented a somewhat devious (if laborious) maneuver for gaining access to the most popular questions posed to a commercial search engine. Nonetheless, a great deal more knowledge could usefully be mined by looking at the query logs themselves, in their entirety. Barring access to such a rich source of questions, it may prove just as informative to focus on the growing stream of linguistic data produced on micro-blogging sites such as Twitter. These new forms of social media encourage people to speak as they think, and to effectively introspect aloud. Amid this innocuous chatter, computers may find just what they need to think and to introspect for themselves.

Acknowledgments

References

Almuhareb, A. and Poesio, M. (2004) Attribute-Based and Value-Based Clustering: An Evaluation. *In Proceedings Of EMNLP'2004*, pp 158-165.

Barnden, J. A. 2006. Artificial Intelligence, figurative language and cognitive linguistics. G. Kristiansen, M. Achard, R. Dirven, and F. J. Ruiz de Mendoza Ibanez (Eds.), *Cognitive Linguistics: Current Application and Future Perspectives*, 431-459. Berlin: Mouton de Gruyter.

Boden, M. 1994. Creativity: A Framework for Research, Behavioural and Brain Sciences 17(3):558-568.

Budanitsky, A. and Hirst, G. 2006. Evaluating WordNet-based Measures of Lexical Semantic Related- ness. *Computational Linguistics,* 32(1):13-47.

Brants, T. and Franz, A. 2006. *Web 1T 5-gram Version 1*. Linguistic Data Consortium.

Falkenhainer, B., Forbus, K. and Gentner, D. 1989. Structure-Mapping Engine: Algorithm and Examples. *Artificial Intelligence*, 41:1-63.

Fass, D. 1991. Met*: a method for discriminating metonymy and metaphor by computer. *Computational Linguistics*, 17(1):49-90.

Fellbaum, C. ed. 2008. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge.

Gentner, D. 1983, Structure-mapping: A Theoretical Framework. *Cognitive Science* 7:155–170.

Hearst, M. 1992. Automatic acquisition of hyponyms from large text corpora. *In Proc. of the 14th International Conference on Computational Linguistics*, pp 539–545.

Lakoff, G. and Johnson, M. 1980. *Metaphors we live by*. University of Chicago Press.

Martin, J. H. 1990. A Computational Model of Metaphor Interpretation. New York: Academic Press.

Ortony, A. 1979. The role of similarity in similes and metaphors. Ortony, A. (Ed.), *Metaphor and Thought,* Cambridge University Press.

Pasca, M. and Van Durme, B. 2007. What You Seek is What You Get: Extraction of Class Attributes from Query Logs. In *Proc. of IJCAI-07, the 20th International Joint Conference on Artificial Intelligence.*

Shutova, E. 2010. Metaphor Identification Using Verb and Noun Clustering. In *the Proc. of the 23rd International Conference on Computational Linguistics*, 1001-1010.

Turney, P.D. and Littman, M.L. 2005. Corpus-based learning of analogies and semantic relations. Machine Learning 60(1-3):251-278.

Veale, T. and Hao, Y. 2007a. Making Lexical Ontologies Functional and Context-Sensitive. In *proc. of the 46th Ann. Meeting of the Assoc. of Computational Linguistics*.

Veale T. and Hao, Y. 2007b. Comprehending and generating apt metaphors: a web-driven, case-based approach to figurative language. In *Proc. of AAAI'2007, the 22nd national conference on Artificial intelligence*, pp.1471-1476.

Way, E. C. 1991. Knowledge Representation and Metaphor. *Studies in Cognitive systems*. Holland: Kluwer.

Wilks, Y. 1978. Making Preferences More Active. *Artificial Intelligence* 11(3):197-223.