From Conceptual "Mash-ups" to "Bad-ass" Blends:

A Robust Computational Model of Conceptual Blending

Tony Veale
School of Computer Science and Informatics
University College Dublin, Belfield D4, Ireland.
Tony.Veale@UCD.ie

Abstract

Conceptual blending is a cognitive phenomenon whose instances range from the humdrum to the pyrotechnical. Most remarkable of all is the ease with which humans regularly understand and produce complex blends. While this facility will doubtless elude our best efforts at computational modeling for some time to come, there are practical forms of conceptual blending that are amenable to computational exploitation right now. In this paper we introduce the notion of a *conceptual mash-up*, a robust form of blending that allows a computer to creatively re-use and extend its existing common-sense knowledge of a topic. We show also how a repository of such knowledge can be harvested automatically from the web, by targetting the casual questions that we pose to ourselves and to others every day. By acquiring its world knowledge from the questions of others, a computer can eventually learn to pose introspective (and creative) questions of its own.

The Plumbing of Creative Thought

We can think of comparisons as pipes that carry salient information from a source to a target concept. Some pipes are fatter than others, and thus convey more information: think of resonant metaphors or rich analogies that yield deeper meaning the more you look at them. By convention, pipes carry information in one direction only, from source to target. But creativity is no respecter of convention, and creative comparisons are sometimes a two-way affair.

When the actor and writer Ethan Hawke was asked to write a profile of Kris Kristofferson for *Rolling Stone* magazine, Hawke had to create an imaginary star of his own to serve as an apt contemporary comparison. For Hawke, Brad Pitt is as meaningful a comparison as one can make, but even Pitt's star power is but a dim bulb to that of Kristofferson when he shone most brightly in the 1970s. To communicate just how impressive the singer-actor-activist would have seemed to an audience in 1979, Hawke assembled the following Frankenstein-monster from the body of Pitt and other assorted star parts:

> "Imagine if Brad Pitt had written a No. 1 single for Amy Winehouse, was considered among the finest songwriters of his generation, had been a Rhodes scholar, a U.S. Army Airborne Ranger, a boxer, a professional helicopter pilot – and was as politically outspoken as Sean Penn. That's what a motherfuckin' badass Kris Kristofferson was in 1979."

Pitt comes off poorly in the comparison, but this is precisely the point: no contemporary star comes off well, because in Hawke's view, none has the wattage that Kristofferson had in 1979. The awkwardness of the comparison, and the fancifulness of the composite image, serves as a creative meta-description of Kristofferson's achievements. In effect Hawke is saying, "look to what lengths I must go to find a fair comparison for this man without peer". Notice how salient information flows in both directions in this comparison. To create a more rounded comparison, Hawke finds it necessary to mix in a few elements from other stars (such as Sean Penn), and to also burnish Pitt's résumé with elements borrowed from Kristofferson himself. Most of this additional structure is imported literally from the target, as when we are asked to imagine Pitt as a boxer or a helicopter pilot. Other structure is imported in the form of an analogy: while Kristofferson wrote songs for Janis Joplin, Pitt is imagined as a writer for her modern counterpart, Amy Winehouse.

This *Pitt 2.0* doesn't actually exist, of course. Hawke's description is a *conceptual blend* that constructs a whole new source concept in its own counterfactual space. Blending is pervasive in modern culture, and can be seen in everything from cartoons to movies to popular fiction, while the elements of a blend can come from any domain of experience, from classic novels to 140-character tweets to individual words. As defined by the cognitive linguists Gilles Fauconnier and Mark Turner (1998, 2002), conceptual blending combines the smoothness of metaphor with the structural complexity and organizing power of analogy. We can think of blending as a cognitive operation in which conceptual ingredients do not flow in a single direction, but are thoroughly stirred together, to create a new structure with its own emergent meanings.

The *Kristofferson-as-Pitt* blend shows just how complex a conceptual blend can be, while nonetheless remaining intelligible to a reader: when we interpret these constructs,

we are not aware of any special challenge being posed, or of any special machinery being engaged. Nonetheless, this kind of blend poses significant problems for our computers and their current linguistic/cognitive-modelling abilities. In this paper we propose a computional middle-ground, called a *conceptual mash-up*, that captures some of the power and utility of a conceptual blend, but in a form that is practical and robust to implement on a computer. From this starting point we can begin to make progress toward the larger goal of creative computational systems that – to use Hawke's word – can formulate truly *badass* blends of their own.

Creative language is a knowledge-hungry phenomenon. We need knowledge to create or comprehend an analogy, metaphor or blend, while these constructs allow us to bend and stretch our knowledge into new forms and niches. But computers cannot be creative with language unless they first have something that is worth saying creatively, for what use is a poetic voice if one has no opinions or beliefs of one's own that need to be expressed? This current work describes a re-usable resource – a combination of knowledge and of tools for using that knowledge – that can allow other computational systems to form their own novel hypotheses from mashups of common stereotypical beliefs. These hypotheses can be validated in a variety of ways, such as via web search, and then expressed in a concise and perhaps creative lingustic form, such as in poem, metaphor or riddle. The resource, which is available as a public web service called *Metaphor-Eyes*, produces conceptual mash-ups for its input concepts, and returns the resulting knowledge structures in an XML format that can then be used by other computational systems in a modular, distributed fashion. The *Metaphor-Eyes* service is based on an approach to creative introspection first presented in Veale & Li (2011), in which stereotypical beliefs about everyday concepts are acquired from the web, and then blended on demand to create hypotheses about topics that the computer may know little about. We present the main aspects of *Metaphor-Eyes* in the following sections, and show how the service can be called by clients on the web.

Our journey begins in the next section, with a brief overview of relevant computational work in the areas of metaphor and blending. It is our goal to avoid hand-crafted representations, so in the section after that we describe how the system can acquire its own common-sense knowledge from the web, by eavesdropping on the revealing questions that users pose everyday to a search engine like Google. This knowledge provides the basis for conceptual mash-ups, which are constructed by re-purposing web questions to form new instrospective hypotheses about a topic. We also introduce the notion of a *multi-source mash-up*, which allows us to side-step the vexing problem of context and user-intent in the construction of conceptual blends. Finally, an empirical evaluation of these ideas is presented, and the paper concludes with thoughts on future directions.

Related Work and Ideas

We use metaphors and blends 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.

Veale and O'Donoghue (2000) proposed an SMT-based model of conceptual blending that was perhaps the first computational model of the phenomenon. The model, called *Sapper*, addresses many of the problems faced by SME – such as deciding for itself which knowledge is relevant to a blend – but succumbs to others, such as the need for a hand-crafted knowledge base. Pereira (2007) presents an alternative computational model that combines SMT with other computational techniques, such as using genetic algorithms to search the space of possible blends. Pereira's model was applied both to linguistic problems (such as the interpretation of novel noun-noun compounds) and to visual problems, such as the generation of novel monsters/creatures for video games. Nonetheless, Pereira's approach was just as reliant on hand-crafted knowledge. To explore the computational uses of blending without such a reliance on specially-crafted knowledge, Veale (2006) showed how blending theory can be used to understand novel portmanteau words – or "formal" blends – such as "Feminazi" (Feminist + Nazi). This approach, called *Zeitgeist*, automatically harvested and interpreted portmanteau blends from Wikipedia, using only Wikipedia itself and Wordnet (Fellbaum, 1998) as resources.

The availability of large corpora and the Web suggests a means of relieving the knowledge bottleneck that afflicts computational models of metaphor, analogy and blending. 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 how these properties allow word-concepts to be clustered into category structures that replicate the semantic divisions made by a curated resource like WordNet (Fellbaum, 1998). Veale and Hao (2007a,b) 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 (2012), how a body of 4000 stereotypes is 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 the way it operates 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 an implicit source of common-sense knowledge. The challenge we face as computationalists lies in turning this implicit world knowledge into explicit representations. For instance, 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 show in this paper how a common-sense representation that is derived from web questions can be used in a model of conceptual blending. We focus on well-formed questions, found either in the query logs of a search engine or harvested from documents on the web. These questions can be viewed as atomic properties of their topics, but they can also be parsed to yield logical forms for reasoning. We show how, by representing topics via the questions that are asked about them, we can also grow our knowledge-base via blending, by posing these questions introspectively of other topics as well.

"Milking" Knowledge from the Web

Amid the ferment and noise of the Web sit nuggets of stereotypical world knowledge, in forms that can be automatically harvested. 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 conceptual blending.

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

Why do pirates wear eye patches?
Why do pirates hijack vessels?
Why do pirates have wooden legs?

Parsing the 2nd question above, we obtain its logical form:

$$\forall x \ \text{pirate}(x) \rightarrow \exists y \ \text{vessel}(y) \wedge \ \text{hijack}(x, y)$$

A computational system needs a critical mass of such commonsense knowledge before it can be usefully applied to problems such as conceptual blending. Ideally, we could extract a large body of everyday musings from the query logs 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 call *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.

Conceptual "Mash-ups"

Google milking yields these frequent questions about *poets*

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?

Querying 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 poets yet which one is unlikely to find in a dictionary. Now suppose a user asks the system to explore the ramifications of the blend *Philosophers are Poets*: this prompts the system to introspectively ask "*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*" (just 2 results). The goal is not to show that these behaviors are as salient for philosophers as they are for poets, rather that they can be meaningful for philosophers.

We refer to the construct *Philosophers are Poets* as a *conceptual mash-up*, since knowledge about a source, *poet*, has been mashed-up with a given target, *philosopher*, to yield a new knowledge network for the latter. Conceptual mash-ups are a specific kind of conceptual blend, one that is easily constructed via simple computational processes.

To generate a mash-up, the system starts from a given target T and searches for the source concepts $S_1 \ldots S_n$ that might plausibly yield a meaningful blend. A locality assumption limits the scale of the search space for sources, by assuming that T must exhibit a pragmatic similarity to any vehicle S_i. Budanitsky and Hirst (2006) describe a raft of term-similarity measures based on WordNet (Fellbaum, 1998), but what is needed for blending is a generative measure: one that can quantify the similarity of T to S as well as suggest a range of likely S'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 (rather than semantic) neighbor of Y, and vice versa. When identifying blend sources for a topic T, we consider the neighbors of T as candidate sources for a blend. Furthermore, if we consider the 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 source S, and rank the neighbors $S_1 \ldots S_n$ of T by their similarity to T.

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, or even interchangeable. Potential sources for T are ranked by their perceived similarity to T, as described above. Thus, when generating mash-ups for *philosopher*, the most highly ranked sources suggested via the Google 3-grams are: *scholar, epistemologist, ethicist, moralist, naturalist, scientist, doctor, pundit, savant, explorer, intellectual* and *lover*.

Multi-Source Mash-Ups

The problem of finding good sources for a topic T is highly under-constrained, and depends on the contextual goals of the speaker. However, when blending is used for knowledge acquisition, multi-source mash-ups allow us to blend a range of sources into a rich, context-free structure. If $S_1 \ldots S_n$ 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 $S_1 \ldots S_n$, 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, for example, "*why do artists use Macs?*" is more likely to be successfully re-purposed for the target of a blend 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 (as calculated

using a WordNet-based metric; see Seco et al., 2006). Both combine to give a single salience measure $Q_{salience}$ in (1):

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

Note that Q_{count} is always greater than 0, since each question Q must be suggested by at least one neighbor of T. Note also that salience is not a measure of surprise, but of aptness, so the larger Q_{count}, the larger $Q_{salience}$. 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 mash-up, and estimate the likelihood of successful knowledge transfer from one topic to another.

Empirical Evaluation

Our corpus-attested, neighborhood-based approach to similarity does not use WordNet, but is capable of replicating the same semantic divisions made by WordNet. In earlier work, 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, we denote this experimental setup as AP214), and report a cluster purity of **0.85** in replicating the category structures of WordNet. But if the 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 purity/accuracy of **0.934** is achieved on AP214. Using 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.

The locality assumption underlying this notion of a pragmatic neighborhood constrains the number of sources that can contribute to a multi-source mash-up. Knowledge of a source S can be transferred to topic T only if S and T are 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 is transferred via mash-ups to all of their neighbors. For instance, "*why do artists use Macs?*" suggests "*musicians use Macs*" as a hypothesis because *artists* and musicians are close neighbors, semantically (in WordNet) and pragmatically (in the Google n-grams); this hypothesis is in turn validated by 5,700 web hits. In total, 410,000 hypotheses are generated from these 10 test terms, and when posed as web queries to validate their content, approx. 90,000 (21%) are validated by usage in web texts.

Just as knowledge tends to cluster into pragmatic neighborhoods, hypotheses likewise tend to be validated in clusters. As shown in Figure 1, the probability that a hypothesis is valid for a topic T grows with the number of neighbors of T for which it is known to be valid (Q_{count}).

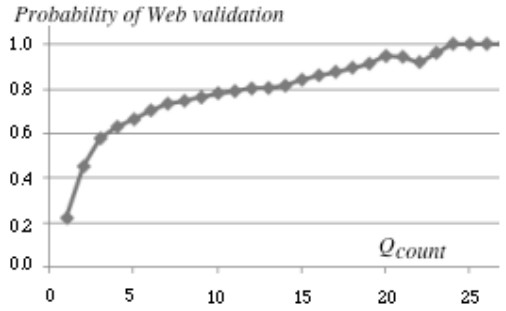

Figure 1. *Likelihood of a hypothesis in a mash-up being validated via web search (y-axis) for hypotheses that are suggested by Q_{count} neighbors (x-axis).*

Unsurprisingly, close 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 ($Q_{salience}$).

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 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.

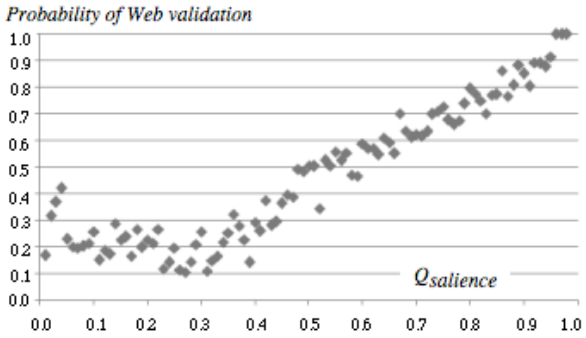

Figure 2. *Likelihood of a hypothesis in a mash-up being validated via web search (y-axis) for hypotheses with a particular $Q_{salience}$ measure (x-axis).*

Figure 3 shows the average rate of web validation for the top-ranked hypotheses (ranked by 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, 85% of the top 20 hypotheses in each mash-up are

validated on by web search as plausible, while just 1 in 4 of the top 60 hypotheses in a mashup is not web-validated.

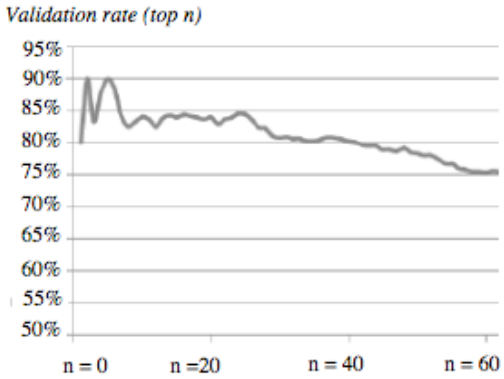

Figure 3. *Average % of top-n hypotheses in a mash-up (as ranked by $Q_{salience}$) that are validated by Web search.*

Figures 1 – 3 show that the system is capable of extracting knowledge from the web which can be successfully transferred to neighboring terms via metaphors and mash-ups, 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.

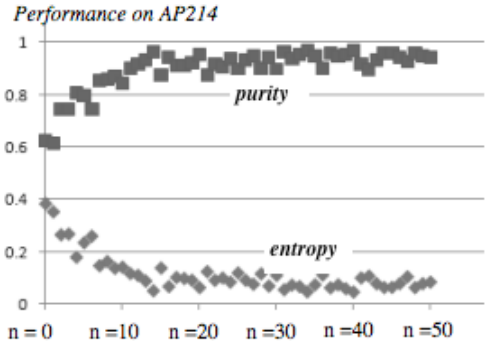

Figure 4. *Performance on AP214 improves as knowledge is transferred from the **n** closest neighbors of a term.*

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 corpus-attested neighbors, so that each term is represented as a conceptual 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. Blends really do bolster our knowledge of a topic with insights that are relevant to categorization.

Conclusions: A Metaphor-Eye to the Future

We have shown here how common questions on the web can provide the world knowledge needed to drive a robust, if limited, form of blending called *conceptual mash-ups*. 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 of multiple concepts.

The web is central to this approach: not only are questions harvested from the web (e.g., via Google "milking"), but newly-formed hypotheses are validated by means of simple web queries. The approach is practical, robust and quantifiable, and uses an explicit knowledge representation that can be acquired on demand for a given topic. Most importantly, the approach makes a virtue of blending, and argues that we should view blending not as a problem of language but as a *tool* of creative thinking.

The ideas described here have been computationally realized in a web application called *Metaphor-Eyes*. Figure 5 overleaf provides a snapshot of the system in action. The user enters a query – in this case the provocative assertion "*Google is a cult*" – and the system provides an interpretation based on a mash-up of its knowledge of the source (cults) and of the target (Google). Two kinds of knowledge are used to provide the interpretation of Figure 5. The first is common-sense knowledge of cults, of the kind that we expect most adults to possess. This knowledge includes widely-held stereotypical beliefs such as that cults are lead by gurus, that they worship gods and enforce beliefs, and that they recruit new members, especially celebrities, which often act as apologists for the cult. The system possesses no stereotypical beliefs about Google, but using the Google 2-grams (somewhat ironically, in this case), it can find linguistic evidence for the notions of a *Google guru*, a *Google god* and a *Google apologist*. The corresponding stereotypical beliefs about cults are then projected into the new blend space of *Google-as-a-cult*.

Metaphor-Eyes derives a certain robustness from its somewhat superficial treatment of blends as mash-ups. In essence, the system manipulates conceptual-level objects (ideas, blends) by using language-level objects (strings, phrases, collocations) as proxies: a combination at the concept-level is deemed to make sense if a corresponding combination at the language-level can be found in a corpus (or in the Google n-grams). As such, any creativity

exhibited by the system is often facile or glib. Because the system looks for conceptual novelty in the veneer of surface language, it follows in the path of humour systems that attempt to generate interesting semantic phenomena by operating at the punning level of words and their sounds.

We have thus delivered on just one half of the promise of our title. While conceptual mash-ups are something a computer can handle with relative ease, "bad-ass" blends of the kind discussed in the introduction still lie far beyond our computational reach. Nonetheless, we believe the former provides a solid foundation for development of the tools and techniques that are needed to achieve the latter. Several areas of future research suggest themselves in this regard, and one that appears most promising at present is the use of mash-ups in the generation of poetry. The tight integration of surface-form and meaning that is expected in poetry means this is a domain in which a computer can serendipitously allow itself to be guided by the possibilities of word combination while simultaneously exploring the corresponding idea combinations at a deeper level. Indeed, the superficiality of mash-ups makes them ideally suited to the surface-driven exploration of deeper levels of meaning.

Metaphor-Eyes should thus be seen as a community resource thru which the basic powers of creative introspection (as first described in Veale & Li, 2011) can be made available to a wide variety of third-party computational systems. In this regard, *Metaphor-Eyes* is a single instance of what will hopefully become an established trend in the maturing field of computational creativity: the commonplace sharing of resources and tools, perhaps as a distributed network of web-services, that will promote a wider cross-fertilization of ideas in our field. The integration of diverse services and components will in turn facilitate the construction of systems with an array of creative qualities. Only by pooling resources in this way can we hope to go beyond single-note systems and produce the impressive multi-note "badass blends" of the title.

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.

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

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

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

Gilles Fauconnier and Mark Turner. (1998). Conceptual Integration Networks. *Cognitive Science,* 22(2):133–187.

Gilles Fauconnier and Mark Turner. (2002). *The Way We Think. Conceptual Blending and the Mind's Hidden Complexities.* Basic Books.

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.

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

Pereira, F. C. 2007. *Creativity and artificial intelligence: a conceptual blending approach*. Walter de Gruyter.

Seco, N., Veale, T. and Hayes, J. 2004. An Intrinsic Information Content Metric for Semantic Similarity in WordNet. In the proceedings of ECAI 2004, the 16th European Conference on Artificial Intelligence. Valencia, Spain. John Wiley

Shutova, E. 2010. Metaphor Identification Using Verb and Noun Clustering. *In Proceedings 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 D. O'Donoghue. (2000). Computation and Blending. *Cognitive Linguistics*, 11(3-4):253-281.

Veale, T. 2006. Tracking the Lexical Zeitgeist with Wikipedia and WordNet. *Proceedings of ECAI-2006, the 17th European Conference on Artificial Intelligence*.

Veale, T. and Hao, Y. 2007a. Making Lexical Ontologies Functional and Context-Sensitive. In *Proceedings of the 46th Ann. Meeting of 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 *Proceedings of AAAI'2007, the 22nd national conference on Artificial intelligence*, pp.1471-1476.

Veale, T. and Li, G. 2011. Creative Introspection and Knowledge Acquisition: Learning about the world thru introspective questions and exploratory metaphors. In Proc. of AAAI'2011, the 25th Conference of the Association for the Advancement of Artificial Intelligence.

Veale, T. 2012 *Exploding the Creativity Myth: The Computational Foundations of Linguistic Creativity*. London: Bloomsbury/Continuum.

Figure 5. *A screen-shot from the computational system* Metaphor-Eyes*, which implements the model described in this paper.* Metaphor-Eyes *shows how we can use conceptual mash-ups to explore what-ifs and to stimulate human creativity. (Note: Because the system has no prior ontological knowledge about Google, each entry above shows a default score of 100 and a support/similarity measure of 0). Please visit* http://Afflatus.UCD.ie *to interact with the* Metaphor-Eyes *system for yourself, or to find out more about the system's XML functionality.*