The Concept Game: Better Commonsense Knowledge Extraction by Combining Text Mining and a Game with a Purpose

Amaç Herdağdelen
CIMEC, University of Trento
Corso Bettini 31, Rovereto, Italy
amac@herdagdelen.com

Marco Baroni
CIMEC, University of Trento
Corso Bettini 31, Rovereto, Italy
marco.baroni@unitn.it

Abstract

Common sense collection has long been an important subfield of AI. This paper introduces a combined architecture for commonsense harvesting by text mining and a game with a purpose. The text miner module uses a seed set of known facts (sampled from ConceptNet) as training data and produces candidate commonsense facts mined from corpora. The game module taps humans' knowledge about the world by letting them play a simple slot-machine-like game. The proposed system allows us to collect significantly better commonsense facts than the state-of-the-art text miner alone, as shown experimentally for 5 rather different types of commonsense relations.

1 Introduction

In order to display human-like intelligence, advanced computational systems should have access to the vast network of generic facts about the world that humans possess and that is known as *commonsense* knowledge (books have pages, grocery has a price...). Developers of AI applications have long been aware of this, and, for decades, they have invested in the laborious and expensive manual creation of commonsense knowledge repositories, such as the Cyc database (Lenat 1995). Thanks to the development of techniques for knowledge-poor induction of knowledge from text, coupled with the sudden massive availability of text on the Web, it has recently become possible to extract literally millions of facts in a matter of hours by Web text mining. However, the knowledge obtained in this way is very noisy. Banko et al. estimate that about 20% of the millions of generic facts they extracted from the Web with a state-of-the-art large scale information extraction system are wrong (Banko et al. 2007).

A different approach – which is based on human computation – to harvesting knowledge from the Web is to get surfers to actively provide the knowledge we need. The Open Mind Common Sense project (Speer 2007) relies on the good will of Web surfing volunteers, while a recent and promising alternative to volunteer work is that of *games with a purpose* (Von Ahn 2006), inducing Web surfers to contribute various kinds of useful knowledge while playing and having fun.

In this study, we propose an architecture that combines a large scale text mining algorithm which outputs candidate commonsense assertions with a Facebook slot-machine-like game that lets players validate those assertions while they play. With respect to a purely text-mining-based system, human validation provides less noisy data. Besides their inherent value, such cleaned-up data can be used to assess the quality of text mining and fed back to the algorithm as labeled training materials. With respect to a purely game-based system where players have to enter statements rather than checking them, we exploit the virtually limitless amount of text-mined data to develop a fast-paced routine that adds to the entertainment value while allowing us to collect more data.

The main goal of the current paper is to introduce our combined architecture, and to show, in a controlled setting, that it allows us to collect significantly better commonsense facts than the state-of-the-art text miner alone. After reviewing some related work in section 2, in section 3 we describe our text mining algorithm, while in section 4 we introduce the game. In section 5, we describe our experimental procedure and report the results. Section 6 concludes the paper with main achievements and future directions.

2 Related work

To the best of our knowledge, the Concept Game is the first system integrating commonsense harvesting by text mining and a game with a purpose. Text mining is performed with BagPack. Like other relation extraction methods (Pantel and Pennacchiotti 2006; Riloff and Jones 1999; Turney 2008), BagPack learns from examples of pairs instantiating a relation and a corpus about the contexts that typically surround the example pairs, and uses these contexts to find new pairs that instantiate the relation. Importantly for the current purposes, BagPack has been shown to achieve state-of-the-art performance in various knowledge extraction tasks, including extraction of ConceptNet-like commonsense tuples (Herdağdelen and Baroni 2009). The text miner *per se* is not the focus of our current work, and any reasonable substitute of BagPack would do. In particular, since we focus on expanding ConceptNet, future experiments could rely on ConceptMiner (Eslick 2006), a system explicitly aimed at ConceptNet expansion. Another possible candidate is the KNEXT (KNowledge EXtraction from Text) system proposed by Schubert and collaborators for extracting "general world knowledge from miscel-

laneous texts, including fiction" (Schubert and Tong 2003; Gordon, Van Durme, and Schubert 2010).

Various games with a purpose collect commonsense knowledge in verbal format. Some, like Verbosity (Von Ahn, Kedia, and Blum 2006) and Common Consensus (Lieberman, Smith, and Teeters 2007), differ from the Concept Game in that they require users to *enter* snippets of knowledge, rather than simply verifying them. Both Verbosity and Common Consensus have been used to populate ConceptNet, the same commonsense resource we aim for. The only other commonsense game we are aware of that does not ask the users to produce statements is the CYC project's FACTory game (*http://game.cyc.com/*). The FACTory's commonsense statements are generated from the CYC repository, and players must tell whether they think the statements are true or false (plus nonsense and don't-know options). Extra points are awarded when a player agrees with the majority answer for a fact and a certain consensus threshold has been reached. The Concept Game differs from FACTory in that it is designed as a fast-paced game where players are rewarded for speed and penalized for wrong statements. Moreover, the purpose of our game is not to verify the knowledge in an existing commonsense resource, but to expand the repository by filtering text-mined facts (and, in the future, to bootstrap new text mining seeds in order to build a self-extending high-quality knowledge base). The Learner system (Chklovski 2003) is also similar to the Concept Game in that it asks users to verify assertions that are generated in a bootstrapping process seeded by a ConceptNet-related knowledge base. However, Learner is not cast as a game, and it harvests new assertions by analogical reasoning over the collected assertions, rather than from raw text.

There are other attempts to crowdsource commonsense data evaluation/collection, by utilizing services like Amazon's Mechanical Turk (*http://www.mturk.com/*), e.g., Gordon et al. (2010). We consider these payment-based crowdsourcing methods as complementing the games-with-a-purpose approach rather than competing with it. For the evaluation of small datasets, crowdsourcing may be a convenient alternative, but as the amount of data to be annotated increases so does the cost of annotation. In contrast, the operational costs of games are usually almost constant (e.g., a small monthly reward to motivate players) and enlarging the user base would not incur any additional costs.

3 The BagPack algorithm

The central idea in BagPack (**Bag**-of-words representation of **Pa**ired **c**oncept **k**nowledge) is to construct a vector-based representation of a pair of terms in such a way that the vector represents both the contexts where the two terms co-occur (that should be very informative about their relation) and the contexts where the single terms occur on their own (possibly less directly informative but also less sparse). BagPack constructs three different sub-vectors, one for the first term (recording frequency of sentence-internal co-occurrence of the first term with basis items which may be unigrams or n-grams depending on implementation), one for the second (with the same kind of information), and one for the co-occurring pair (keeping track of the basis items that occur in sentences where both terms occur). The concatenation of these three sub-vectors is the final vector that represents the pair. BagPack is a supervised algorithm: The vectors constructed in this way from labeled example pairs (including positive and negative examples of the target relation) and a corpus are fed to a Support Vector Machine that is then used to classify or to rank unlabeled pairs according to their confidence scores (co-occurrence counts are logarithmically transformed before training the SVM). In the experiments reported below, we use a C-SVM implementation (*http://asi.insa-rouen.fr/enseignants/ arakotom/toolbox/*) with a linear kernel and the cost C set to 1.

Herdağdelen and Baroni test BagPack extensively, reporting results comparable to state-of-the-art algorithms on TOEFL synonym detection, answering SAT analogy questions and modeling verb selectional preferences, plus promising preliminary results on the extraction of ConceptNet-like commonsense assertions (Herdağdelen and Baroni 2009).

4 The Concept Game

The Concept Game (CG) is a game with the purpose to collect common sense from laypeople. It is based on the idea that production of verbal information is a significant burden on the player and it is possible to design enjoyable games without requiring the players to produce assertions. During the game, the players do not try to produce commonsense assertions but they verify already collected candidate assertions. CG is presented in the context of a slot machine which produces random assertions. A meaningful assertion is a winning configuration. The trick is that the winning configurations do not dispense rewards automatically, but first they have to be recognized by the player to "claim their money". In this way, players tell us which assertions they found meaningful.

The game consists of independent play sessions each of which starts with an allocation of 40 seconds. First, the player sees three slots with images of rolling reels. They correspond to left concept, relation, and right concept of an assertion. Then, the contents of the slots are fixed one by one with some values picked from the database and as a result an assertion is displayed. At that point, the player has to press one of two buttons labeled as "Meaningless" or "Meaningful". If the player correctly identifies – and claims points for – a meaningful assertion (s)he is rewarded with two points and two bonus seconds (i.e., true positives are rewarded). If the player claims money for a meaningless assertion, (s)he loses three points and three seconds (i.e., false positives are penalized). However, pressing the meaningless button does not change the score or the remaining time (i.e., neither false negatives are penalized nor true negatives are rewarded). The feedback is conveyed to the player visually and acoustically (e.g., in case of a reward a green color flashes, in case of a penalty a red color flashes). The reels roll again, and the process repeats. This continues until the end of the allocated time, which can get longer or shorter depending on rewards and penalties.

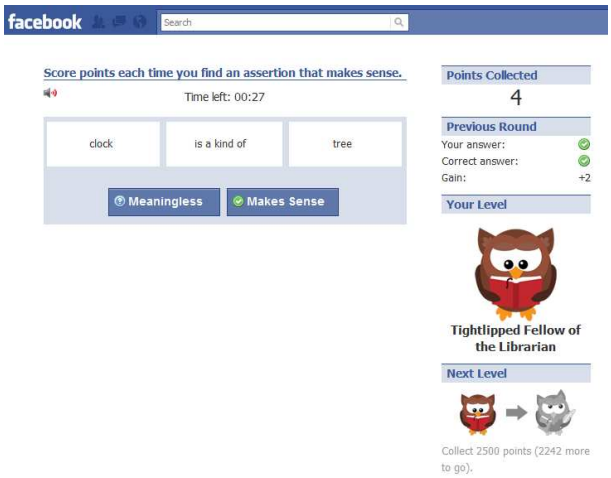

Figure 1: Screenshot of a playing session in Concept Game

In the previous description, we pretended that the game already knows which labels are meaningful, and rewards or penalizes the user accordingly. This is not the case. In CG, we employ a validation policy similar to the honor-based, proof-of-payment method employed in many public transportation systems. In such a system, instead of checking every user, periodic controls are carried out to make sure the abuse of the system is effectively discouraged. As long as the cost of the penalty is high enough, the game does not have to check the validity of all responses of a player. For validation, the game displays candidate assertions whose actual labels are known (i.e., they are from the training set of the BagPack) and for all other cases the game presents candidates whose labels are unknown and accepts and gives rewards for all meaningful responses for such assertions. It is indeed this latter set of responses that enables the game to collect common sense from the players. Suitable choices of rewards and penalties combined with proper controls will force rational players who try to maximize their scores to play honest. In current implementation, the probabilities of showing a meaningless, a meaningful, and a candidate assertion are 0.4, 0.3, and 0.3 respectively. In other words, 30% of the collected responses are for the candidate assertions proposed by BagPack.

CG is implemented in Python and published as a Facebook application. It was fun to design and implement, and we believe that playing it is also fun. A typical screenshot of the game is given in Figure 1.

Technically, the game is almost equivalent to asking a group of raters to tick those assertions from a list which they think make sense. This is a dull task especially if there are few meaningful assertions compared to meaningless ones. In the context of a slot machine, however, the experience of seeing many meaningless assertions becomes part of the game, which creates an expectation in the player that (hopefully) resolves with a "winning" configuration. The relatively short session timing, combined with the need to be accurate because wrong claims are penalized, should keep the attention level of the players up, and consequently add to the fun. We made sure that players are aware of their achievements (they see total and session scores they have collected) and have an incentive to keep playing (we also display a top score list that shows the users who scored highest in a single session). That said, this study is focused on the efficiency and reliability of CG rather than its fun aspects and we prefer having more direct empirical evidence (e.g., number of users playing it on a daily basis) once it is fully public before speculating more about the issue.

5 Experiments

Our series of experiments start by sampling a seed assertion set from ConceptNet and end by outputting a set of assertions mined from Wikipedia with high likelihoods of being instantiations for the given relations. ConceptNet (*http://conceptnet.media.mit.edu/*) is a freely available semantic network consisting of relationships between concept pairs. Of more than 25 relation types in ConceptNet 4, we focus on five that represent rather different ways in which concepts are connected, correspond to more (IsA) or less (SymbolOf) traditional ontological relations, and tend to link words/phrases from different syntactic classes: IsA (*cake, dessert*); AtLocation (*cake, oven*); HasProperty (*dessert, sweet*); MotivatedByGoal (*bake cake, eat*); SymbolOf (*Sacher Torte, Vienna*).

For clarity of discussion, we present our pipeline in four steps. In the first step, we collected seed instances from ConceptNet and had expert raters annotate them. In the second step, we built and trained BagPack models on the resulting datasets. In the third step, we mined a large number of candidate assertions from a corpus and ranked them according to the confidence values we got from BagPack. The top scoring ones were passed as input to the fourth and last step, in which the players of CG scored them. We also had expert raters annotate the candidate assertions presented to the players. We evaluated the performance of BagPack and CG by using the raters' answers as golden standard. Below, we will explain each step in detail.

Seed collection from ConceptNet

For each relation, we randomly sampled a set of approximately 250 assertions from ConceptNet (SymbolOf is instantiated by 151 assertions only, and we used all of them). In addition to the original ConceptNet assertions, we artificially constructed an equal number of bogus assertions by randomly picking an assertion and changing i) either one of its associated concepts with a random concept from ConceptNet (*Sacher Torte SymbolOf win election*), ii) or the original relation with another of the five relations we work with (*Sacher Torte IsA Vienna*).

We recruited a total of 22 expert raters, all advanced students or researchers in artificial intelligence, semantics or related fields. The raters were given precise instructions on the purpose of the procedure and had to annotate assertions as *meaningful* or *meaningless*. For each rater, we computed the probability of agreement with the majority vote on a random assertion and, as a precaution to ensure high-quality data, we discarded the responses of five raters with a proba-

Relation	Meaningful	Meaningless	Total
AtLocation	196	320	516
IsA	206	337	543
HasProperty	228	356	584
MotivatedByGoal	216	339	555
SymbolOf	107	79	186

Table 1: Meaningful and meaningless assertion decomposition of ConceptNet-based training datasets.

Relation	Precision (O)	Precision (A)
AtLocation	0.68	0.15
IsA	0.63	0.17
HasProperty	0.59	0.23
MotivatedByGoal	0.69	0.15
SymbolOf	0.69	0.09

Table 2: Precision measures for original (O) and artificially generated (A) assertions, based on expert rating.

bility lower than 0.70. The mean and median number of annotated assertions across the remaining 17 raters were 427 and 260 respectively. Only the 2,384 assertions rated by at least two raters were considered for further analysis. The final label of an assertion was decided by the majority vote and the ties were broken in favor of meaningfulness. Table 1 summarizes the annotation results for each relation. The annotated assertions served us as training datasets in the next step. Note that some of the original assertions coming from ConceptNet were rated as meaningless (for example: *bread IsA put butter*; *praise IsA good job*; *read newspaper MotivatedByGoal study bridge*). These assertions should serve as high-quality negative instances given that they made their way into ConceptNet at one time as plausible assertions.

A by-product of this experiment is an evaluation of the quality of the assertions in ConceptNet. If we limit ourselves only to the responses given for the original assertions in ConceptNet we can get an estimate for the precision of the assertions in ConceptNet. In a similar fashion, the (artificially generated) random assertions can provide a baseline performance. In Table 2, we list the ratio of the number of assertions annotated as meaningful to the total number of assertions (i.e., precision) for each relation for the original and artificial subsets. The overall precision of our sample from ConceptNet is around 0.65.

Training BagPack on Common Sense

For each of the five datasets coming from the previous step, we trained a separate BagPack model. The co-occurrence vectors were extracted from the Web-derived English Wikipedia and ukWaC corpora, about 2.8 billion tokens in total (we use the pre-processed versions from *http://wacky.sslmit.unibo.it*). Since ConceptNet concepts are often expressed by multiple words (*Sacher Torte*, *eat too much*, ...), we employed a shallow search for the concept phrases. Basically, for a single phrase, we looked for the occurrence of the constituents with possible intermittent extra elements where the concept phrase spans no more than the twice of its original length. For two phrases, we required that the constituents do not overlap and do not span more than a 20-word-range taken together. For efficiency reasons, a maximum of 3,000 sentences were used to extract co-occurrence statistics for a given pair. The subvectors for each of the components of a pair were populated by their (log transformed) sentence-internal co-occurrence counts with (lemmatized) unigrams. In order to fit the dataset into available memory, we used as features (vector dimensions) only those unigrams that co-occur with at least 10% of the instances in a given dataset, resulting in approximately 25,000 selected features per dataset.

Relation	AUC	95% Confidence Interval
AtLocation	0.72	(0.69-0.76)
IsA	0.68	(0.64-0.71)
HasProperty	0.70	(0.58-0.76)
MotivatedByGoal	0.55	(0.49-0.60)
SymbolOf	0.67	(0.58-0.76)

Table 3: BagPack AUC on five ConceptNet relations, by 10-fold cross validation on the training set.

Table 3 reports areas under the ROC curves (AUC) and associated confidence intervals for 10-fold cross validation of the BagPack procedure on the training data. The area under the ROC curve can be interpreted as the probability of a random positive instance having a higher confidence score than a random negative instance (Fawcett 2006). An AUC value of 0.5 means chance performance and for all relations, the performance of BagPack was significantly above that level. However, AUC for MotivatedByGoal was barely above chance level and even the best AUC performance of 0.72 that was obtained on AtLocation was quite low, suggesting that BagPack alone cannot be used to extract reliable commonsense assertions from corpora.

Common Sense Mining

We mined the dependency parsed Wikipedia corpus made available by the WaCky project (see link above). The top 10,000 most frequent verbs, nouns and adjectives were considered potential concept heads, and we extracted potential concept phrases with a simple dependency grammar aimed at spotting (the content words of) noun, verb and adjective phrases (for example, the grammar accepts structures like `Adj Noun`, `Verb Adj Noun` and `Adv Adj`). In this phase, we were not interested in the semantic association between the concept pairs but simply tried to generate lots of pairs to feed into the BagPack models trained in the previous step. Once all heads were identified in a sentence, we combinatorially populated the concept phrases by following the set of allowed dependency links in the grammar that matched the sentence parse. The last step was to identify all pairs of potential concept phrases that do not overlap in the sentence. We repeated this for all sentences and kept track of the co-occurrence frequencies of the concept pairs. For an example sentence: "The quick brown fox jumped over the lazy dog.", some possible concept pairs that might be extracted are: (*fox, dog*), (*quick fox, dog*), (*brown fox, lazy dog*), (*fox, jump*), (*fox, jump over lazy dog*). The combinatorial aspect of the phrase generation process resulted in

many concept pairs (up to hundreds for a single sentence), that we pruned in various ways. We did not allow concept phrases with more than 6 words or containing numerals. We computed pointwise mutual information (PMI) of the pairs (Church and Hanks 1990) and kept only the pairs with a PMI higher than 4 (in natural logarithm) and which contained at least one concept with a frequency higher than 40,000.

The pair extraction algorithm applied to Wikipedia produced 116,382 concept pairs. Then, we randomly sampled 5,000 pairs (containing 5,385 unique concept phrases) from this set, and generated 10,000 directed pairs by ordering them in both directions. Approximately 68% of the concepts in the sampled pairs were single words, 30% were 2-word phrases, 2% contained 3 or more words. Some example concept phrases that were mined are *wing*, *league*, *sport team*, *fairy tail*, *receive bachelor degree*, *write several book*, *father's death*, and *score goal national team*.

Next, we associated the sample pairs with each of the five relations we study, obtaining a set of assertions that contain the same concept pairs, but linked by different relations. As a preliminary analysis, two expert raters annotated 1,000 randomly sampled assertions from this set. The ratio of assertions that were annotated as meaningful by at least one rater was estimated to be 0.11. This low ratio justified the idea of scoring the assertions with the trained BagPack models in order to rank them and pick candidates, rather than blindly feeding all of them into the game.

We extracted BagPack-style vectors for the sampled concept pairs. The BagPack vectors were scored by the trained BagPack models for each of the 5 relations, and the triples formed by the concept pairs and the relations were ranked by the resulting BagPack confidence scores.

Approximately 400 triples at the top of the BagPack ranked lists for each relation (over 2,000 triples in total) were annotated by two expert raters in order to provide a golden standard for further analysis. The raters' overall Cohen's kappa was 0.37. The raters agreed about 183 meaningful assertions (8.8% of the full set) and 1,508 meaningless assertions (72.6%). Any assertion that was annotated as meaningful by at least one rater was assumed to be meaningful for purposes of assessing the players' performance.

Game in Action

In the current stage of game development, we wanted to experiment with a small group of users, rather than letting the game out in the open. We asked 18 people by email to play the game, mostly college students and staff that the authors personally knew. Unlike the raters used in the previous steps, players were *not* experts in AI, semantics or related fields. The game was open to this "semi-public" for approximately 10 days.

In total, 25 players (7 presumably invited by the ones we contacted) responded and provided approximately 5,000 responses. The ratio of players who scored an assertion as meaningful was the *CG score* of the assertion. In addition to CG scores, we already had the BagPack confidence scores of the assertions and the PMI values for the associated pairs. CG, BagPack and PMI can be seen as three different methods to rank candidate pairs. BagPack acts as a baseline for CG while PMI is a (relatively) trivial baseline.

Relation	Meaningful	Total
AtLocation	139	383
IsA	84	322
HasProperty	128	355
MotivatedByGoal	128	406
SymbolOf	90	349

Table 4: Summary of the Wikipedia-based datasets used in game evaluation. Labels are based on rater annotation.

Relation	CG	BagPack	PMI
AtLocation	0.77	0.58	0.44
IsA	0.77	0.62	0.54
HasProperty	0.68	0.55	0.55
MotivatedByGoal	0.72	0.57	0.53
SymbolOf	0.71	0.67	0.56

Table 5: Area under the ROC curve (AUC) on candidate assertion set.

In our analysis, we considered the 1,815 assertions which were scored by at least two players, split across relations as shown in Table 4. The assertions that were labeled as meaningful consisted of 798 unique concepts of which 164 (21%) were not attested in the entire ConceptNet (not just the sample we used as initial seed).

Using the expert raters' judgments from section 5 as the gold standard, we computed relation-specific ROC curves for the CG, BagPack, and PMI scores. The areas under the ROC curves are given in Table 5.

In addition, in Table 6 we report the recall values of the three models for each relation at the precision values we estimated for the original ConceptNet (see section 5). These recall values are rough estimates of what we can get from the sample candidate set if we want to keep the precision level at what we already have in ConceptNet (which seems a reasonable ad interim aim). With the exception of SymbolOf, the output of CG was able to match the precision of our ConceptNet sample while retaining considerable recall values. For three of the five relations, BagPack either could not achieve comparable precision values (MotivatedByGoal) or achieved such values only marginally (i.e., AtLocation, IsA). We repeated the same experiments by optimizing the BagPack's SVM cost parameter with cross-validation but did not observe any significant increase in AUC.

Relation (precision)	CG	BagPack
AtLocation (0.68)	0.50	0.01
IsA (0.63)	0.50	0.01
HasProperty (0.59)	0.47	0.22
MotivatedByGoal (0.69)	0.36	0
SymbolOf (0.69)	0	0.10

Table 6: Recall of models at the estimated ConceptNet relation-specific precision value.

6 Conclusion

From a corpus, we extracted millions of concept pairs which are likely to be involved in a semantic relation and narrowed down this set to a mere hundred thousand by a combination of PMI and frequency filters. This set is meant to be used as the source of candidate assertions to be evaluated first by BagPack and then CG. The fact that the game was not publicly available at the time of this study, and thus we only had access to a small player base, forced us to sample 5,000 undirected pairs out of this set to get preliminary results. We used a gold standard of expert human raters' judgments and contrasted it with the predictions obtained from CG and BagPack alone. In all cases our text mining algorithm outperformed simple PMI-based ranking, and the game-based validation brought a considerable improvement in the quality of the top assertions. We were also able to provide evidence that text mining introduces considerable variety into the concept set that the assertions are built upon (21% new concepts with respect to ConceptNet).

We thus laid the premises for a synergy that combines the advantages of text mining (quantity, low cost) and manual resource creation (quality); and this does not even take into account the further improvements that should derive from re-training the text miner on the larger training data sets created by the game. Increasing the precision of the text miner would also help us to display more unknown assertions without a significant impact on the game experience (i.e. players would still be able to score points without the support of the meaningful assertions we display).

Concept Game is currently fully functional and open to public as a Facebook application (*http://apps.facebook.com/conceptgame/*). In the short term, we are looking for ways to make the game more attractive to a wider non-specialized audience. We would like to convert the lemma sequences produced by BagPack into natural sounding sentences. We have recently started to offer small gifts to top players as an incentive to start and keep playing. Once we gain a reasonably wide player-base and construct a dataset of commonsense assertions, we plan to share the dataset publicly.

Another interesting possibility for future analysis is to look for cultural differences in assertions that receive contrasting ratings from players from different continents. Using Facebook as our platform allows us to access demographics of players for statistical analysis purposes.

While these and many other avenues of development and analysis should be pursued, we believe that our current results make a strong case for the feasibility of an approach that mixes text mining and social intelligence to harvest commonsense knowledge on a large scale.

References

Banko, M.; Cafarella, M.; Soderland, S.; Broadhead, M.; and Etzioni, O. 2007. Open information extraction from the web. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2670–2676.

Chklovski, T. 2003. *Using Analogy to Acquire Commonsense Knowledge from Human Contributors*. Phd thesis, MIT, Cambridge, MA.

Church, K., and Hanks, P. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics* 16(1):22–29.

Eslick, I. 2006. *Searching for Commonsense*. Ms thesis, MIT, Cambridge, MA.

Fawcett, T. 2006. An introduction to roc analysis. *Pattern Recogn. Lett.* 27(8):861–874.

Gordon, J.; Durme, B. V.; and Schubert, L. 2010. Evaluation of commonsense knowledge with mechanical turk. In *In NAACL Workshop on Creating Speech and Language Data With Amazons Mechanical Turk*.

Gordon, J.; Van Durme, B.; and Schubert, L. 2010. Learning from the web: Extracting general world knowledge from noisy text. In *Proceedings of the AAAI 2010 Workshop on Collaboratively-built Knowledge Sources and Artificial Intelligence*. ACM.

Herdağdelen, A., and Baroni, M. 2009. BagPack: A general framework to represent semantic relations. In *Proceedings of the EACL GEMS Workshop*, 33–40.

Lenat, D. 1995. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM* 11:33–38.

Lieberman, H.; Smith, D.; and Teeters, A. 2007. Common consensus: a web-based game for collecting commonsense goals. In *Proceedings of IUI07*.

Pantel, P., and Pennacchiotti, M. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of COLING-ACL*, 113–120.

Riloff, E., and Jones, R. 1999. Learning dictionaries for information extraction by multilevel bootstrapping. In *Proceedings of the National Conference on Artificial Intelligence*, 474–479.

Schubert, L., and Tong, M. 2003. Extracting and evaluating general world knowledge from the Brown corpus. In *Proceedings of the HLT-NAACL 2003 workshop on Text meaning-Volume 9*, 7–13. Association for Computational Linguistics Morristown, NJ, USA.

Speer, R. 2007. Open Mind Commons: An inquisitive approach to learning common sense. In *Proceedings of the Workshop on Common Sense and Intelligent User Interfaces*.

Turney, P. 2008. A uniform approach to analogies, synonyms, antonyms and associations. In *Proceedings of COLING*, 905–912.

Von Ahn, L.; Kedia, M.; and Blum, M. 2006. Verbosity: A game for collecting common-sense knowledge. In *Proceedings of CHI*, 75–78.

Von Ahn, L. 2006. Games with a purpose. *Computer* 29(6):92–94.