Generation of Educational Content through Gameplay

Adam Skory, Maxine Eskenazi

Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, U.S.A.

askory@cs.cmu.edu, max@cs.cmu.edu

Abstract

Language learning games require large quantities of content to be educational. This paper presents a boot-strapping method to generate game content using a combination of existing digital resources and the human computation of game players themselves. It describes a baseline method for hint generation to be used in a two-player game in which players select hints to assist each other. It then shows that the data gathered from player actions contains significant agreement in evaluation of automatically generated content. The quality of this data is verified against data from tasked-based crowdsourcing. Results demonstrate that this type of data may be used to extend and improve the hint generation method.

Index Terms: educational games, language learning, crowdsourcing

1. Introduction

This paper describes a method to generate hints as content usable in a game involving phrasal verbs (PVs). By not relying on manual authoring of hints by experts, cost may be reduced and the amount of content significantly increased, thus enabling students to play for longer.

1.1. Generating game content

For a game to teach PVs, an extensive amount of high quality content is needed: a set of target PVs, sentences containing those PVs, and hints for the PVs in the context of these sentences. It is possible to produce sentences and collocations of reasonable quality using automatic methods [1]. However for every target PV presented in such a game, many more words are needed to serve as possible hints. A full-scale game deployment may teach hundreds of PVs, necessitating thousands of hints.

To produce a large number of hints, we started with a manually-tuned baseline method for hint generation. We hypothesized that this method produces hints of sufficient quality for students to be able to play the game. We also hypothesized that significant agreement can be found when extracting human evaluations of hint quality from gameplay logs. This data could be used to sort and expand upon the baseline set of potential hints for each target PV.

1.2. Teaching phrasal verbs

PVs are a class of collocations prominent in English. Collocations are frequently occurring word sequences with meaning or usage patterns beyond the regular generative rules of syntax or semantics. Collocations are difficult for students because knowledge of the semantics of a collocation's constituents is not sufficient to determine its meaning as a phrase. Our partners at the English Language Institute of the University of Pittsburgh include PVs in their curriculum. Since educational game content should be relevant to students' coursework [2], we designed a game focused on PVs.

1.3. Hints as near-synonyms

The concept of *hints*, in the context of this game, is defined as cues that one player chooses to help the other player find a missing phrase in a cloze sentence. For example, for "blow up" in the sentence "The pothole will blow up my tire", one could use "explode" and "detonate". The ability of a student to retrieve "blow up" from memory is a function of both the contextual cues available and her knowledge of that target phrase [3]. At the most general level, the constituents of word knowledge can be broken down into *form*, including phonetics and orthography; *meaning*, referring to concepts and referents; and *use*, encompassing the word's syntax [4].

Multiple reasons suggest focusing only on *meaning* based hints. Many of the target PVs are composed of common words. If spelling and pronunciation of these words are known to a student, she will have no trouble with the *form* of the verb phrase. However she will not necessarily be able to predict the meaning and usage of the phrase - by the definition of collocation. In this game, both students see a cloze sentence with the target PV removed, so a *use* cue is already provided. Furthermore, research in vocabulary acquisition shows the importance of inter-word associations in terms of similarities in *meaning* rather than in terms of similarities in *form* [3,4,5].

With a focus on *meaning* hints, one important semantic relation to discuss is synonymy. Absolute synonymy requires two words to share the same meaning in all possible contexts. This is a restrictive definition; more appropriate for this task is the broader concept of *near-synonymy*. That is, the measure of the degree of meaning overlap shared between two words. This can be defined as the number of contexts shared in which meaning is consistent within a chosen level of specificity [6]. For this game a relatively low level of specificity of meaning may still provide semantic cues for lexical retrieval. These observations drive the baseline hint generation method described in Section 2.

1.4. Related work

Some techniques used to generate distractors could be used as a starting point for hint generation. One criterion of distractor generation applicable to hints is that meaning should be similar. A constraint placed on distractors, that is not placed on hints, is that they should not be acceptable in place of a target [7]. Thus methods used to generate semantically similar distractors, without the latter constraint, may serve as the basis for hint generation. [7] present a method to generate distractors using corpus statistics. They search for words that co-occur frequently with either the context before or the context after the target word, but, importantly, do not co-occur frequently with the context *both* before *and* after the target. A generalization of this method, used in the statistical hint generation method below, is to remove this final constraint and retrieve words that frequently co-occur with the context before *and* after the target. Other work [8,9] generates distractors by choosing words of the same part of speech (POS), similar frequency in a corpus, and, again, low co-occurrence frequency with the context of the target. These methods do not offer a suitable generalization to influence the baseline method because they do not directly reflect the criterion of related meaning.

2. Hint Generation: The Baseline

The baseline approach to hint generation takes advantage of three resources: WordNet, Thesaurus.com, and the English GigaWord corpus. The method ranks words as hints for each target phrase according to a hint-score based on a linear combination of manually-tuned scores from each resource.

For each PV found in WordNet [10], synonyms, hyponyms, and hypernyms can be extracted. Hyponyms are near-synonyms with a specificity of meaning lower than the target. Less specific words tend to be more frequent, and so are most likely to be known by learners of English [4]. We consider these the most likely to be good hints. Hypernyms, on the other hand, may be too specific in meaning compared to the target PV. Therefore, they will be near-synonyms of the target when their meaning matches the correct context. Hypernyms are the least likely to be good hints. The likelihood that absolute synonyms are good hints is considered to be in between that of hyponyms and hypernyms:

$$h_{WordNet}(w,t) = \begin{cases} 1.1 & \text{if } hyponym(w,t) \\ 1.0 & \text{if } synonym(w,t) \\ 0.9 & \text{if } hypernym(w,t) \end{cases} \quad (1)$$

We also extracted near-synonyms from Thesaurus.com. Unlike WordNet, this thesaurus does not label the type of semantic relation between each word and the target, so we uniformly assign words from the thesaurus a more conservative hint-score:

$$h_{thsaurus}(w,t) = 0.8 \quad (2)$$

Finally, we use a concordancing algorithm to extract hints from the English GigaWord corpus [11], as a generalization of [7]'s method for generating distractors. For a given PV, the algorithm records all two-word contexts in the data. Next it finds all the words that also occur in any of these two-word contexts. A word's statistical hint-score is calculated as the number of contexts it shares with the target PV, normalized by the total number of contexts for that phrase.

For a target sequence t we define a context c as a pair $c=(w_1,w_2)$ such that the ordered super-sequence (w_1,t,w_2) is found at least once in the corpus. We can then define C_t as the set of all c for sequence t in the corpus. Finally, the statistical hint-score $h(w,t)$ for any word w and target t is calculated as:

$$h_{stat}(w,t) = \frac{|C_t \cap C_w|}{|C_t|} \quad (3)$$

The final hint-score for (w,t) is the sum:

$$h(w,t) = h_{WordNet}(w,t) + h_{thesaurus}(w,t) + h_{stat}(w,t) \quad (4)$$

Preliminary experiments showed that using WordNet and the thesaurus alone enabled us to generate tens of hints for each of a set of 100 common PVs. The statistical algorithm generated hundreds of results with a large distribution of scores. Manual inspection of the top 50 results for the same 100 PVs showed that a small number of automatically generated hints were usable. This group of hints was not, however, consistently highly ranked. We hypothesize that harnessing the evaluations of non-expert game players will provide ranking data to improve this baseline method.

3. Games for gathering data

Crowdsourcing is generally understood to mean the use of a "crowd" of non-experts to perform small, repeatable tasks that are difficult to automate; labeling of images or transcribing noisy speech, for example. One common form of crowdsourcing is the use of micro-task markets such as Amazon's Mechanical Turk (AMT). In this model, workers are motivated by small payments to complete many short tasks. Using human computation games (HCG) rather than payments to gather data is another possibility. To date, HCG have been used to gather labels for images, to tag music, to perform word sense disambiguation, and more [12].

3.1. Gathering hint evaluations with HCG

The combination of CALL games and HCG is not new. For example, Gruenstein et al. [13] developed a flashcard-based game to collect non-expert audio transcriptions. Our work combines HCG with CALL gaming in a novel way: the data collected by our game can be used to directly improve its own content.

In two-player HCGs, data is elicited by controlling the interactions between two players [14]. Motivated by the competitive challenge of the game, players often find effective ways to get higher scores that circumvent the original design, at the cost of data quality [12]. For example, in the case of a game in which one player can freely type hints to prompt another player to guess, steps must be taken to ensure that the hints given are not too easy - obvious misspellings of the target word, for example - rendering the game trivial. Restricting the prompting player's role to that of clicking on a pre-defined set of potential hints avoids this problem. Players do have the option to type in additional hint suggestions, but only at the end of every round. These suggested words are not shown directly to other players, but are recorded for later analysis.

3.2. The design of Hint Hunting

Hint Hunting is a game in which players practice PVs while providing hint evaluations. Students log in to a website to find partners for gameplay. During one game session, the players take turns playing one of two roles: the *guesser* or the *hint hunter*. Both players see a cloze sentence, but only the *hint hunter* knows the missing PV. *The guesser*'s job is to guess the missing phrase, while the *hint hunter*'s job is to help the *guesser* do so before a timer runs out. During each round words gradually appear synchronously on both players' screens. Some of these words are good hints, and some are not. If the *hint hunter* clicks on a word, the word will remain on both screens. Words on which the *hint hunter* does not click will shrink and disappear. By "hunting" the good hints, the *hint hunter* can help the *guesser* find the missing word or phrase. When the timer runs out, the following round begins with a new sentence, and the players' roles are reversed. For the purposes of this study, *Hint Hunting* is implemented only as an HCG interface for data gathering; the educational potential of these game mechanics is left for future work to evaluate.

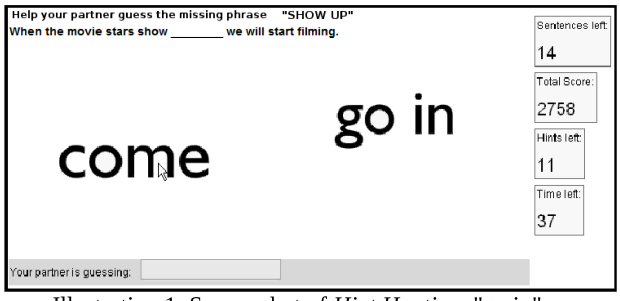

Illustration 1: Screen shot of *Hint Hunting*, "go in" and "come" are chosen as hints.

4. Implementation

4.1. Presentation of content

For the present study, 20 target PVs were randomly split into four sets of five each. At the start of a game session, one of the four sets of targets is chosen randomly. Players have the option of playing several sessions, but are not able to play the same set of targets twice. A player who has seen all four sets is blocked from further play. Also, if two players do not share any unseen sets, they will not be able to play with each other.

At the beginning of every round both players are given 10 seconds to read the sentence before the timer starts. At that point the timer lasts 50 seconds. While the timer is running, potential hints appear at the rate of one every three seconds. As soon as a hint enters the screen it shrinks and disappears within about four seconds if it is not chosen by the *hint hunter*.

One full session of the game consists of 5 PV targets with three rounds per target. Every round has a different sentence and different hints. In a full session players will encounter 5

PVs, 15 sentences, and up to 195 potential hints. The relative order of PV targets was randomized as was presentation of sentences and hints.

4.2. Other content generation

A candidate set of PVs was generated using a POS tagger and mutual information scores for all pairs within a set of 300 common verbs and 70 prepositions. We then used our baseline method to extract hints for every PV in this set, and ranked the PVs according to the cumulative score of their top 40 hints. Based on this ranking, we chose the top 20 PVs. These 20, according to the baseline, have the highest number of predicted good hints.

The cloze sentences used in the game were extracted from the GigaWord corpus and filtered for quality based on the co-occurrence scores of the words in the sentence with each PV [1]. From the resulting list of candidate sentences three were chosen manually to control for length, formatting, similarity, and word-sense.

4.3. The study

For this study we made the game publicly available online and recruited non-native English speakers through social media and on AMT. All players were asked their native language and their level of expertise with English. To prevent native English (L1) speakers from representing themselves as non-natives, AMT players were asked to translate one of several randomly chosen sentences into their native language. Data from players who were unable to do so was rejected. While L1 speakers could use a web service to translate into a non-native language, the extra time required to do so should deter this.

5. Results

Over the course of one week, 102 unique, non-native English speakers played at least one round of *Hint Hunting*. 508 rounds were completed by pairs of these players. The mean and median number of rounds completed by each user was 10. Feedback from users indicated that some players recruited through AMT left the system before finishing a full session.

For the purposes of gathering human evaluation of the hints generated by the baseline method, two types of data are the most important: hint-clicks and hint-inputs. 628 hint-clicks were gathered, as were 656 hint-inputs.

5.1. Hint-clicks as a function of time

The average round length was only 26 seconds, meaning, on average, the *hint hunter* only had opportunities to click on at most 5 hints before the *guesser* found the solution. In comparison, there were 13 such opportunities when the *guesser* was not able to find the solution before the timer ran out. This happened in 186 rounds, or 37% of the time. Looking in more detail, we find a standard deviation of more than 5 seconds for average round length per PV. For example, players were able to guess 'blow up' within 13 seconds but needed 39 seconds to guess 'knock off.' In terms of hints seen, this is an important difference. Taking into account the initial ten seconds before hints appear, this means that the *hint hunter* only would have *one* opportunity to click on a hint before the round ended for 'blow up', but *nine* opportunities to click on hints for 'knock off.' Figure 1 shows the distribution of target PVs by number of total hint-clicks.

5.2. Hint-clicks as a function of players

Significant trends in the click patterns are evident after normalizing for total click numbers. Figure 2 shows the average proportional distribution of clicks for all PV targets. The x-axis corresponds to a minimum proportion of clicks received; a value of 0.05 here means that at least 5% of all hint-clicks for a PV target went to a specific hint. The y-axis is the number of hints that received at least that proportion,

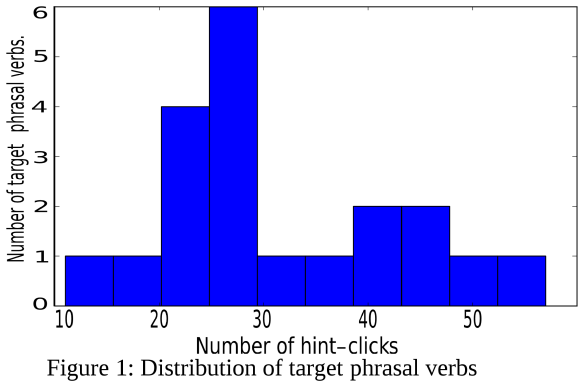

Figure 1: Distribution of target phrasal verbs according to number of total hint-clicks.

averaged over 20 targets. The error bars, corresponding to one standard deviation, demonstrate that the number of hints receiving the largest proportion of clicks is fairly stable, while the number of hints receiving just a small proportion of clicks is more variable.

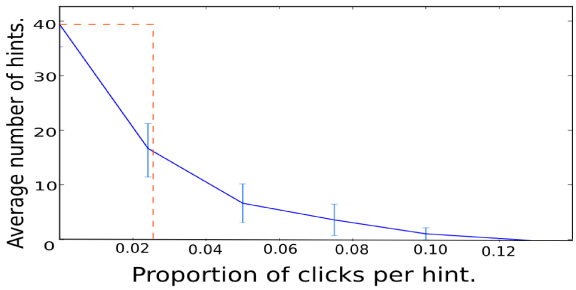

Figure 2: Distribution of hints by minimum proportion of clicks (solid) and expected distribution if probability of clicks is uniform (dashed).

Manual inspection showed that hints agreed upon by at least 6% of the players were 'good' hints. Of 780 total hints selected using the baseline method, 124 were 'good' by this definition, or 16% of the total. Of the remaining 84%, those that are not good hints still contribute to game mechanics by acting as foils. The significance of agreement on the best hints can be shown according to its deviation from a uniform distribution of hint-clicks, corresponding to the null hypothesis that all players clicked randomly. Chi-square tests of the distributions of hint-clicks for all PVs reveal a statistically significant mean *p-value* of 0.0036. Inspection of highly agreed-upon hints shows that they have appropriate meaning and level. For example, the most preferred hints for *get along* are: *relate*, *get on*, and *succeed*, while words with inappropriate meaning, such as *highway*, or inappropriate level, such as *flourish*, received zero hint-clicks.

The data also reveals patterns in hint-clicks in terms of differences among players. Because not all players played the same number of rounds, we normalize by number of rounds played and consider the distribution of hint-clicks per player per round. 71% of players fall below the average value of 0.66 hint-clicks per round, and only 29% above it. One prodigious player averaged 7 hint-clicks per round, while 41 players never clicked on any hints at all. We investigate the relationship of this difference to the amount of time a player's partner needed to guess the phrase. The number of hint-clicks per person per round is correlated with the average amount of time spent playing as the *hint hunter*, with a statistically significant coefficient of 0.63 ($p < 0.001$).

5.3. Hint-inputs

All players were given the opportunity to type in additional hint suggestions after every round. Players did so 56% of the time overall, and each time 1.5 hints were typed in on average. If the same hint is input by several people, this is an indication that it is a good hint. The number of hints input

more than once was 105, out of 464 unique typed-in hints. The average number of times a unique hint was input was 1.4, with a standard deviation of 1.1. “Steal”, as a hint for “rip off”, was the most popular: it was input 11 times.

At any given point in the game it is possible that a player has not yet seen many of the potential hints. If that player types in a baseline hint before seeing it in the game, this is strong evidence that it is a good hint. This occurred a total of 74 times for 33 unique hints. Of the 464 unique hints typed in, 363 were not generated by the baseline method.

5.4. Validation using task-based crowdsourcing

The significant agreements found within the distribution of hint-clicks by non-expert game players can be validated using data from task-based crowdsourcing. We ran one task on AMT for L2 speakers of English - in order to obtain data from a population of non-experts similar to the population of game players - and another task for L1 speakers of English. Both tasks were identical. Workers were provided with the target PV, the same three sentences seen in the game, and a list of all baseline hints plus all typed-in hints, in random orders. Workers were asked to choose all good hints for the target PV in the context of those sentences. These tasks were broken up by PV target, and workers could do as few as one, or as many as 20. The average number of PVs reviewed by L2 workers was 4.3, and for L1 workers, 3.9. For each PV, L2 workers labeled 6.8 hints, while L1s labeled 9.3. The set of hints for each PV was evaluated by 29 L1s, but only 23 L2s.

The L2s gave an average of 179 total hint-labels to the hints for each PV, and the L1s gave 298. To test how well results from the game-based data match the task-based data, each hint's proportion of hint-labels can be compared to its proportion of hint-clicks by plotting each as a function of the other. L1 hint-labels correlate overall with a coefficient of 0.26 ($p < 0.001$), and L2 labels correlate with a coefficient of 0.22 ($p < 0.001$). Hint-clicks agreed with hint-labels to a much higher degree for several PVs; evaluations for “take off” correlated with coefficients of 0.66 ($p = 0.0007$) and 0.57 ($p < 0.05$) for L1 and L2 workers respectively.

After manual inspection we found that the hints chosen by at least 50% of L2 AMT workers were 'good'. Since there were 20 PVs, each with 39 hints selected by the baseline method, the total number of hints we examined was 780. On average, AMT workers found 7.3 good hints for each PV, thus 146 hints, or 19% of the total were 'good'. Of these 146 hints, 101 were also found to be 'good' by game players, showing that crowdsourcing and the HCG method agreed 69% of the time according to these criteria.

Both L1 and L2 workers significantly preferred hints that were typed in during gameplay. The mean number of positive hint-labels by L1 workers for these hints was 6.2, while the mean for hints only produced by the baseline method was 4.5. A *t-test* of the two samples gives a *p-value* less than 0.001 for this difference. For L2 workers these means were 4.2 and 2.4 respectively, with a difference *p-value* less than 0.001.

6. Discussion

The results show that significant patterns can be extracted from the evaluations of hints made by non-expert, L2 game players. These patterns also correspond significantly to evaluations from task-based crowdsourcing. In the process we have discovered, however, that balancing the demands of data gathering with the demands of game mechanics is difficult. For this study, content presentation and player pairing were randomized. This led to more instances than expected of players guessing a target very quickly. In these cases our game design resulted in a negative effect on the amount of data gathered. Furthermore, the variance in the data gathered for each of the 20 PVs in this study suggests that future work will need to test a larger number phrases to show the generalizability of this method. Additional work will investigate the use of student and content models to better match students with content of the appropriate level. However, despite sensitivity to the difficulty of content, our HCG approach still delivered significant agreement on a subset of hints for all 20 PVs. For HCG to produce this agreement without student and content models our hypotheses are clearly reinforced.

The results of this study provide evidence that it is possible to seed educational game content with automatic methods that are “good enough” for play, and then control player interactions to elicit meaningful evaluations of the quality of this content. In future work we will test learn-to-rank algorithms with these evaluations as labels, and quantify their ability to improve the hint generation method for a larger number of PVs.

7. Conclusion

We have described a method to automatically generate hints for PVs for an educational game. A study in which non-native speakers of English played the game revealed that the distributions of hint-clicks and hint-inputs by players indicate significant agreement on the quality of a subset of the hints. These results were confirmed with human evaluations using crowdsourcing with both non-native and native speakers of English. Future work will apply this data to iteratively improve our hint generation model.

8. Acknowledgments

We would like to thank friends and colleagues who participated in play-testing *Hint Hunting*. This project is supported by REAP-PT, and funded by the Information and Communication Technologies Institute (ICTI).

9. References

[1] A. Skory and M. Eskenazi, “Predicting Cloze Task Quality for Vocabulary Training,” Proc. BEA, NAACL, Los Angeles, 2010.

[2] F.W.M. Yip and A.C.M. Kwan, “Online vocabulary games as a tool for teaching and learning English vocabulary,” Media, vol. 43, 2006, pp. 233-249.

[3] C.A. Perfetti and L. Hart, “The lexical quality hypothesis,” Precursors of functional literacy, C. Elbro, L. Vehoeven, and P. Reitsma, eds., John Benjamins, 2002, pp. 189-213.

[4] I.S.P. Nation, Learning Vocabulary in Another Language. I.S.P. Nation, Cambridge University Press, 2001.

[5] J. Milton, Measuring Second Language Vocabulary Acquisition, Bristol: Multilingual Matters, 2009.

[6] P. Edmonds, “Semantic Representations of Near-Synonyms for Automatic Lexical Choice,” 1999.

[7] J. Lee and S. Seneff, “Automatic Generation of Cloze Items for Prepositions,” Interspeech, 2007.

[8] C.-L. Liu, C.-H. Wang, Z.-M. Gao, and S.-M. Huang, “Applications of Lexical Information for Algorithmically Composing Multiple-Choice Cloze Items,” Proc. BEA, ACL, Ann Arbor, 2005

[9] J.C. Brown and M. Eskenazi, “Automatic Question Generation for Vocabulary Assessment,” Computational Linguistics, 2005.

[10] C. Fellbaum, ed., WordNet, Cambridge, Massachusetts: MIT Press, 1998.

[11] R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda, “English Gigaword Fourth Edition,” 2009.

[12] C. Ho, T. Chang, J. Lee, J.Y. Hsu, and K. Chen, “KissKissBan : A Competitive Human Computation Game for Image Annotation,” KDD-HCOMP, Paris, France: 2009.

[13] A. Gruenstein, I. Mcgraw, and A. Sutherland, “A self-transcribing Speech Corpus: Collecting Continuous Speech with an Online Educational Game,” SLaTE, 2009.

[14] L. von Ahn and L. Dabbish, “Designing Games with a Purpose,” Comm. ACM, vol. 51, Aug. 2008, pp. 58-68.