Towards Automatic Identification of Effective Clues for Team Word-Guessing Games

Eli Pincus & David Traum
USC Institute for Creative Technologies
12015 Waterfront Dr
Playa Vista, CA 90094, USA
pincus,traum @ict.usc.edu

Abstract

Team word-guessing games where one player, the clue-giver, gives clues attempting to elicit a target-word from another player, the receiver, are a popular form of entertainment and also used for educational purposes. Creating an engaging computational agent capable of emulating a talented human clue-giver in a timed word-guessing game depends on the ability to provide effective clues (clues able to elicit a correct guess from a human receiver). There are many available web resources and databases that can be mined for the raw material for clues for target-words; however, a large number of those clues are unlikely to be able to elicit a correct guess from a human guesser. In this paper, we propose a method for automatically filtering a clue corpus for effective clues for an arbitrary target-word from a larger set of potential clues, using machine learning on a set of features of the clues, including point-wise mutual information between a clue's constituent words and a clue's target-word. The results of the experiments significantly improve the average clue quality over previous approaches, and bring quality rates in-line with measures of human clue quality derived from a corpus of human-human interactions. The paper also introduces the data used to develop this method; audio recordings of people making guesses after having heard the clues being spoken by a synthesized voice (Pincus and Traum, 2016).

Keywords: Word-Guessing Games, Clue-Giver, Clue, Guess, Point-Wise Mutual Information

1. Introduction

Word-guessing games have a long history as a popular form of entertainment and have been created in many different forms. One kind of game is *challenge games*, where a moderator or opponent provides clues as a challenge to one or more guesser(s). Some forms of challenge games include crossword puzzles, (first published in the newspaper, *New York World* in 1913), hangman, and twenty-questions, "college bowl" tournaments, and the TV show *Jeopardy*. Guessing games of this sort have also been used for pedagogical purposes.

Another type is *team games*, where the giver and receiver are both on the same team (possibly in competition with other teams), and the goal of the clue giver is not to challenge the receiver(s) but to efficiently allow the receiver to guess the target. Generally, the clues are not allowed to contain any form of the target word the clue-giver is attempting to elicit and there are time restrictions on how long the receiver has to make a correct guess. Parlor games such as Taboo and Catch-Phrase are examples of team-games. People even enjoy watching others play these more interactive versions as evidenced by the popularity of television shows like Password and Pyramid. The fast-paced interactive nature of the dialogue that takes place in these games is challenging for today's dialogue systems to emulate. Skilled human clue-givers are able to rapidly select an effective clue (a clue capable of eliciting a correct guess) for an arbitrary target word taking into account previous (as well as overlapping) receiver guesses. Skilled human clue-givers are also able to optimize their turn-taking policy in order to minimize the time it takes for the receiver to say a correct guess by giving new clues at appropriate times (possibly interrupting the receiver.)

There are a number of web resources and databases that can be mined for the raw material for clue generation such as dictionary.com, wikipedia, learnersdictionary.com, freebase.com, WordNet etc. Many of the clues derived from these sources are of low quality and very unlikely to elicit a correct guess from a human guesser. This motivates the need for an automatic method to curate clues collected from these resources for the high-quality clues more likely to elicit a correct guess. In this paper, we develop an automatic method to estimate the effectiveness of clues, resulting in a clue corpus that is more comparable to human-generated clues in terms of an average guessability than prior work. The method uses textual features (from the clues in the corpus) that are predictive of a clue's effectiveness. Our proposed method could be used by agents to come closer to emulating a skilled human giver's ability to rapidly select effective clues. The paper also introduces the data used to develop this method; audio recordings of people making guesses after having heard the clues being spoken by a synthesized voice.

In the next section, we describe previous work related to automated word-guessing game players and automatic clue generation. In section 3. we describe the data used in our experiments. In section 4. we describe the experiment conducted in order to create a gold-standard for clue-effectiveness. In section 5. we describe our experimental method, including baseline and human-estimated metrics for clue effectiveness, and a machine-learning experiment for predicting clue effectiveness using clue-features. In section 6. we present the results, showing that the automated clue pruning method results in a set that is far superior to randomly selected clues, and closely approximates human-generated clue quality. Section 7. offers some discussion on our experimental designs, results, and possible future directions. Finally, in section 8. we present our conclusions.

2. Previous Work

There are a number of dialogue agents that can engage in guessing games with a user. The website `http://www.20q.net` allows users to play the challenge game twenty questions, with a computer guesser. The site serves as the front-end interface to a neural network trained to play the twenty questions game originally developed and patented by Robert Burgener (Burgener, 2006). (Sawaki et al., 2008) describe a system that provides clues for a user to guess the name of a famous person, using descriptions generated from encyclopedia articles. Clues are ordered for presentation to generate progressively easier hints. Experiments on clue ranking were performed by (Higashinaka et al., 2007). This work is somewhat similar to our current work in that clues were automatically evaluated for difficulty, however it has some important differences. The agent in (Sawaki et al., 2008) was presenting a challenge to a user, for pedagogical purposes, and thus the first clues given should be difficult. By contrast, the agent in (Pincus et al., 2014) acts as a teammate in a collaborative game, in which the goal is for the receiver to guess as efficiently as possible. Moreover, (Higashinaka et al., 2007) focused on an ideal ordering of clues rather than the difficulty level of clues themselves.

Our work is most similar to the RDG-Phrase game described in (Paetzel et al., 2014) and (Pincus et al., 2014). In this game, the targets are common words or phrases rather than famous people. A corpus of audio and video recordings of humans playing the RDG-Phrase, a timed word guessing game, is described in (Paetzel et al., 2014). Some examples of targets and human clues from this corpus are shown in Table 1. We use a section of this corpus, and the annotations in (Pincus and Traum, 2014) to estimate the average guessability of human clues. (Pincus et al., 2014) introduced an automated clue-giver and a method of automatically generating clues from online sources. (Pincus et al., 2014) estimated clue quality for clues in its corpus by presenting subjects clues in textual form via a web interface. The interface allowed subjects to type in guesses with no time-constraints. The (Pincus et al., 2014) corpus included some clues that were comparable in guessability to clues given by a human clue-giver. However, that corpus contained a much lower percentage of guessable clues than did the human generated clues in the corpus collected by (Paetzel et al., 2014). Our current work collects a much larger corpus of clues and presents the clues via audio instead of text to human guessers.

Table 1: Human Clue Examples

Target	Clue
Ambulance	"car for an emergency "
Video	"um you a cinematographer shoots this"
Convertible	"the roof comes down on a car its called a"

3. Clue Corpus

This section will discuss the general purpose clue corpus that was created as well a special purpose subset of this clue corpus. The general purpose clue corpus that was created by automatically mining web resources and the WordNet database including how the raw text from these sources were processed to create clues that can be used for gameplay. The special purpose subset of this clue corpus was used in order to evaluate the ability of machine learning models to predict whether or not a clue is effective.

3.1. Corpus

Our corpus is based on the one described in (Pincus et al., 2014), which contains clues sourced from three sources: *WordNet* (Miller, 1995), the *Wikipedia* web-page associated with the target word, as well as the *Dictionary.com* webpage associated with the target word. These clues are generally analogous to types of clues generated by human cluegivers and discussed in (Pincus and Traum, 2014). Queries to *WordNet* for a given word can yield lists of that word's synonyms, antonyms, hypernyms and hyponyms. These types of clues were commonly utilized by human cluegivers in the RDG-Phrase corpus. Another common cluetype used by human clue-givers are **Partial-Phrase** clues which are composed of words frequently used with the target word. **Partial-Phrase** clues could be formed by removing the target word in the example sentences scraped from the target word's *Dictionary.com* page. The first sentence in a word's *Wikipedia.com* web-page serve as one source of the most common clue-types utilized by human clue-givers, **DescriptionDef** clues which describes or defines the target word. Examples of automatically generated clues, their associated target-word, their type, and their source can be found in Table 2 (note some of these clues would likely be effective in game-play while others would not). The specific clue types for the general purpose clue corpus were named as follows. If a type of clue starts with *wn* this indicates the clue was generated from *WordNet*. *WordNet* clue type names are composed of a POS and some form of the other words that were used in the original query[1] to *WordNet*. For example, *wnVerbSyn* indicates that the original query to *WordNet* used to generate this clue requested synonyms of the *Verb WordNet* senses of the target word. For more information on the type of relations available in WordNet refer to (Miller, 1995). If the name of a type of clue does not start with *wn* and is not *wiki*, which refers to clues composed of the first sentence of the target word's *Wikipedia* page, then the clue was generated from scraping the target word's *Dictionary.com* page. In this case the type name is generally an abbreviation for the section of the Dictionary.com that the clue was scraped from. For instance, *syn* clues were scraped from the synonyms section of the target word's *Dictionary.com* page.

We generated a new general purpose clue corpus that can be used for game-play using the same sources discussed above for 978 common nouns obtained from a list of common nouns found on the internet and generated clues from the 3

[1]Wordnet was queried via a java wrapper found at http://lyle.smu.edu/ tspell/jaws/

Table 2: Automatically Generated Clue Examples [2]

Target	Type	Clue	Source
Songs	wnNounHypo	"lullaby *is one type of it*"	WordNet
Apparel	exampSent	"wearing blank includes any costume or article of clothing that people wear"	Dictionary.com
Mass	wnNounDef	"the property of something that is great in magnitude"	WordNet
Suit	wnNounHyper	"*a type of* businessman"	WordNet
Lunch	wiki	"blank is a midday meal of varying size depending on the culture"	Wiki
Trouble	wnVerbSyn	"distract"	WordNet
Zoo	wnNounDef	"the facility where wild animals are housed for exhibition"	WordNet
Owner	def	"to receive what is due to one"	Dictionary.com
Spoon	wiki	"in physics, a blank is disturbance or oscillation , that travels through matter or space, accompanied by a transfer of energy	Wiki
Run	def	"in rapid flight"	Dictionary.com
Writing	wnVerbTropos	"impress"	WordNet
Operation	wnUsageExample	"they paid taxes on every stage of the blank"	WordNet
Oil	wnNounHypo	"crude"	WordNet
Fruit	exampSent	"during many summer days, all the blank flies needed for experiments die from the heat."	Dictionary.com
Woman	wiki	"a blank is a female human"	Wiki

sources mentioned above, creating a total corpus of 171,660 clues. The average number of clues per target is just over 175 (compared to 39 in (Pincus et al., 2014)). There are 109,662 (63.9%) clues from *WordNet*, 61,265 (35.7%) clues from *Dictionary.com*, and 733 (0.004%) clues from *Wikipedia*.

3.2. Clue-Processing

All of the raw clues obtained in this corpus were preprocessed in two ways. First, the utility Lexical Variants Generation (Lvg)[3] was utilized to replace the target word and any of its inflected forms in the clue text with the word *"blank"*. 61,727 (36.0%) of the clues in this corpus contain the word *"blank"*. Second, if the clue type could be broadly categorized as hypernym, hyponym, or antonym text was either prepended or appended to the raw clue in order to make the clue more explicitly lead a receiver to the target word. In the case of hypernym *"a type of"* was prepended to the clue. For example, a hypernym clue for the target word "dog" was *"domestic animal"* which became *"a type of domestic animal"*. The text *"is one type of it"* was appended to hyponym clues. For instance, for the target word *"dog"* the hyponym clue *"dalmation"* became *"dalmation is one type of it."* Finally for antonym clues the text *"the opposite of"* was prepended to the clue text. An antonym clue *"bottom"* for the target word *"top"* thus became *"the opposite of top"*.

4. Gold Standard

The special purpose clue corpus was used in a mechanical turk experiment where the clues in this subset were spoken by a Text-To-Speech system and audio recordings of turker guesses were collected and annotated for whether they contained the target (Pincus and Traum, 2016). We refer to this experiment as the gold standard experiment for clue effectiveness; and we refer to this special purpose corpus as the gold standard corpus for clue effectiveness.

One obvious measure of a clue's effectiveness is whether or not the clue is able to elicit a correct guess from a human receiver in a reasonable amount of time. We propose to make this attribute the gold standard for effective clue classification.

In order to obtain this information for a subset of the clue corpus described in Section 3. we directed mechanical turk recruited participants to a web application we developed that elicited spoken guesses. **Participants** were required to be native english speakers, have 92% HIT approval ratings or higher, and have completed at least 100 prior HITs. Unlike (Pincus et al., 2014), we presented the clues to turkers using spoken language and imposed a time limit on the turkers to respond with guesses. Recordings of the text-to-speech system NeoSpeech's James [4] speaking 317 different automatically generated clues were played to different turkers over the web. The frequency of the different clue types in this special purpose corpus can be found in Table 3. The turkers were instructed to make as many guesses as they could once the audio clue started playing. The recording of guesses for each clue ended 6 seconds after the audio containing the spoken clue stopped playing and a pop-up window appeared informing the turker of the clue's target-word. The experiment was designed to play a sequence of 30 clues to each turker followed by a final recording asking for a test-task to be completed ("say the word *strawberry*") to ensure the turker was making a best effort. If this final recording was empty or contained audio other than the word *strawberry*; we did not use that turker's recordings in

[2] text in italics is prepended/appended during clue-processing as discussed in 3.2.

[3] http://nlm.nih.gov/research/umls/new_users/online_learning/LEX_004.htm

[4] http://www.neospeech.com

Table 3: Experiment Corpus Clue Type Freq. Info.

Type	# of clues (% of experiment clue corpus)
wnNounSyn	31 (9.8%)
wnNounDef	30 (9.5%)
def	30 (9.5%)
wnNounHyper	27 (8.5%)
exampSent	27 (8.5%)
wnNounHypo	26 (8.2%)
wnVerbDef	26 (8.2%)
wnVerbSyn	23 (7.3%)
wnVerbHyper	23 (7.2%)
wnVerbUsageExample	20 (6.3%)
wnNounUsageExample	19 (6.0%)
syn	15 (4.7%)
idiomPhrase	10 (3.2%)
wnNounAnt	7 (2.2%)
wiki	3 (0.9%)

our analysis. For unknown reasons many participants began the experiment but did not finish. Incomplete sets of recordings were used in our data analysis only if a subset of the incomplete set passed a manual spot check testing if the recorded guesses seemed to be a best effort.

Clues for 87 different target words including targets like *bomb*, *ornament*, *fowl*, and *breakfast* were used. Only automatically generated clues (as opposed to human generated clues) were used. Multiple clues for the same target were played to different turkers in order to ensure data analysis would be able to differentiate clue-effectiveness from target difficulty; although we leave this for future work. In total 457 recordings of turkers making guesses were recorded. The first author annotated the guess recordings, labeling each recording with a 1 if a correct guess was made and 0 if not. A recording was considered to contain a correct guess even if it was only partially correct (e.g. - a guess of *"paper"* for the target *newspaper*). Table 5 has sample data from the experiment, including one effective clue and one ineffective clue for each of two different targets. It is important to note here that this experiment only provides information on the effectiveness of a single clue (rather than a clue sequence). The experiment was designed in this manner in order to avoid the exponential increase in the amount of data that would be required to determine the effectiveness of an arbitrary clue sequence for an arbitrary target. We leave that for future work.

5. Method

We discuss several different metrics we developed for human-level clue-giving ability as well as a baseline metric for automatic clue-giving ability in order to provide more context to our machine learning experiment results. We perform machine learning experiments in order to determine the predictive value of simple textual features for determining a clue's effectiveness (capability to elicit a correct guess). We use the Weka Machine Learning Library's naive bayes classifier in our experiments (Hall et al., 2009). We perform 10-fold cross validation with our folds stratified across classes.

We introduce some general notation in Equation 1 before discussing the previously mentioned metrics.

$$\begin{gathered} N = total\ \#\ of\ clues\ (given\ or\ in\ corpus) \\ c = total\ \#\ of\ single\ clues \\ leading\ to\ a\ correct\ guess \end{gathered} \tag{1}$$

5.1. Clue-quality comparisons

Baseline We use random clue selection as the baseline for the effective clue prediction task. Random selection here represents a completely naive clue-giving agent that only has the ability to randomly select a clue from the population of automatically generated clues for a given target word. In order to compute the likelihood for random selection selecting an effective clue we simply compute Equation (2).

$$\frac{\#\ clues\ that\ elicit\ corr.\ guess\ from\ turkers}{N} \tag{2}$$

Human-level In order to provide more context for the results of our automatic method it is necessary to consider the effectiveness of clues produced by a typical human clue-giver. Since we are ultimately interested in full RDG-Phrase gameplay, in which multiple clues can be given for a target, we used examples from the corpus from (Paetzel et al., 2014), with annotations from (Pincus and Traum, 2014), identifying clues and correctness of response. However, we need a way of estimating the quality of individual clues given this data, and there is not an obvious scoring methodology to use to discern effective clues (generated by a human clue-giver) from ineffective ones (as any clue given prior to a correct guess can contribute to the receiver's arrival at the correct guess). We have therefore defined three measures to try to approximate clue quality, given appearance in a clue sequence. We include an *upper-bound* measure, a *lower-bound* measure and *expected guessability* measure, that assign different weights to a clue that appears in a successful sequence. The annotations for 8 rounds of the RDG-Phrase game (involving 4 different human clue-givers) that are discussed in (Pincus and Traum, 2014) were used to calculate these statistics.

An *upper-bound* score for a human clue's effectiveness can be arrived at if each clue in a clue sequence leading to a correct or partially correct guess is considered effective and given a score of 1. Implicit in this upper-bound is that each clue could elicit a correct guess from a receiver on its own (an analysis of the RDG-Phrase corpus shows this to be a very generous (unrealistic) assumption -see further comments in section 7. If a target-word is skipped or time runs out before a correct guess is made each clue in that sequence is considered ineffective and receives a score of 0. Using these optimistic assumptions the chance of a human clue-giver generating an effective clue can be calculated by equation (3).

$$\frac{\#\ of\ clues\ part\ of\ a\ correct\ clue\ sequence}{N} \tag{3}$$

A *lower-bound* score for each clue's effectiveness can be computed by giving only clues that elicited correct guesses

without being preceded by additional clues a score of 1 and all clues that were in a sequence of more than one clue a score of 0. This assumes that a clue sequence's effectiveness can be totally attributed to synergies from the combination of clues in the sequence rather than to any single clue's effectiveness (unless of course the clue sequence is of length one). These pessimistic assumptions provide yet another way, shown in equation (4), to compute the likelihood that a human clue-giver's next clue is effective.

$$\frac{c}{N} \tag{4}$$

As a compromise between these extremes, we define an *expected guessability* score for each clue in a sequence leading to a correct or partially correct guess, where partial credit (between the above extremes of 0 and 1) is given for each clue in the sequence. For simplicity, for sequences larger than 1, we define the expected guessability to be $1/(t+1)$ where t represents the total number of clues in the sequence. For single clues, we assign a value of 1, as in both of the above measures. The intuition behind this measure is that the method distributes the credit equally between each clue and a synergistic combination of clues. If a target-word is skipped or time runs out before a correct guess is made each clue in that sequence is considered ineffective and receives a score of 0. Taking the weighted average of the clue's effectiveness scores then provides an alternative measure of how likely a human clue-giver is to generate an effective clue; this calculation can be found in equation (5).

$$\frac{\Big(\sum_{m \in S}\big(\frac{length(m)}{length(m)+1}\big)\Big)+c}{N} \tag{5}$$

$$S=\{all\ clue\ sequences\ leading\ to\ a\ correct\ guess \mid length(sequence)>1\}$$

It is interesting to note that our definitions for human lower-bound, upper-bound, and expected guessability converge to the same value in the case where a correct guess comes after one clue.

5.2. Automatic Clue selection

Table 4: Features Used for Clue Quality Selection

Features
Clue Source
Type of clue + (e.g - wnNounHyp, exampleSentence)
Clue originally contained target-word + (replaced by "blank" during pre-processing)
of words in clue +
Average PMI information +
Max PMI measure +

Feature Selection We perform feature selection using Weka's attribute selection method ChiSquaredAttributeEval which ranks the attributes based on computing an attribute's chi-square statistic with respect to the class. We then use a greedy approach where we start with all attributes and remove the lowest remaining ranked attribute from the ChiSquaredAttributeEval one by one as long as effective clue classification precision is increasing (we discuss why we focus on effective clue precision in 6.).

Features We have extracted some simple textual features from the clues utilized in the mechanical turk experiment. These features are listed in Table 4. A + indicates that this feature is part of the optimal feature set found by our feature selection method. The features include: the clue source (*WordNet*, *Wikipedia*, or *Dictionary.com*), the clue type as discussed in 3., a binary feature of value 1 if the original clue contained the target word otherwise of value 0, as well as Point-wise mutual information (for the words in the clue and the clue's target word) features. The model utilized to calculate the PMI features was built on a corpora containing millions of web blog entries, it is a subset of the spinn3r dataset discussed in (Burton et al., 2009). The point-wise mutual information features for a clue were calculated in two ways. An average PMI for each clue was calculated by taking the average of the average PMI of all constituent clue-words with the target word for the clue, as shown in equation (6), and a max PMI for each clue was calculated by taking the maximum value of equation (6) for all the constituent clue-words. The optimal feature set includes every feature but the clue source. Although the feature set we use in these experiments do not satisfy the assumption of conditional independence made by the Naive Bayes classifier, previous work has shown that the NB classifier has yielded promising results in other text classification tasks even when the features utilized were not completely independent of one another (Dumais et al., 1998). Our results, presented in Section 6., are also consistent with this observation.

$$\frac{PMI(clueWord, target)+PMI(target, clueWord)}{2} \tag{6}$$

6. Results

The results of the machine learning experiment can be found in Table 6. Since the mechanical turk experiment collected data for 317 unique clues and 45 of those clues were able to elicit a correct guess the likelihood that random selection, the baseline method, generates an effective clue is $45/317$ (14.2%). We are most concerned with maximizing the precision (as opposed to recall) of classification for effective clues because successful game play for an arbitrary target-word usually only requires a few effective clues. The precision results reflect the likelihood of the automatic method selecting an effective clue from the clue corpus. The results seen in Table 6 demonstrate that the likelihood of selecting an effective clue is higher for the automatic method than if the baseline random clue selection method is used. The "*" indicates that the difference of the baseline's likelihood and the method's (associated with the asterisk(s)) likelihood is statistically significant[5]. The results from Table 7 show that the likelihood of selecting an effective clue using the automatic method

[5]chi-square test - *: $p = 0.029$

Table 5: Guess Annotation Examples

Target	Clue	Guess	GuessAnnotation Code
Bomb	"An explosive device fused to explode under specific conditions."	"bomb, pressure plate, ..."	1
Bomb	"H-blank is one type of it"	<Silence>	0
Tendency	"a blank to talk too much"	"Tendency"	1
Tendency	"the trend of the stock market"	"up down"	0

Table 6: Baseline & Automatic Method Results

Methodology	**% of Effective Clues**
Baseline	14.2%
Automatic Method	34.6%*

Table 7: Human Clue-Giving Metric Results

Methodology	**% of Effective Clues**
Human (Lower-Bound)	20.0%
Human (Expected-Guessability)	31.8%
Human (Upper-Bound)	79.1%

Table 8: RDG-Phrase Sample Dialogue
Target-Word: *hour*

Player	**Utterance**
Giver	"now"
Receiver	"time now"
Giver	"not minutes"
Receiver	"seconds"
Giver	"not seconds"
Receiver	"hours"
Giver	"okay"

falls within the expected-guessability and upper bounds of human likelihood. Thus a culled corpus pruned using the automatic method with the exhaustive feature selection algorithm should be a much more promising set of clues for examining human-machine gameplay, where a system like Mr Clue in (Pincus et al., 2014) provides automatically generated spoken clues to a human receiver.

7. Discussion

In the future, we may want to revise the way in which clue effectiveness is measured. In Section 4. we gave a clue an effectiveness score of 1 if any turker was able to guess the clue within the time limit. We could refine this to look at the percentage of correct guesses that are elicited, and also how fast they are received (which is important for the final game, because the faster targets are guessed, the more targets can be attempted and more points can be accumulated within the time limit). Also, the annotation counted something as successful even if was only partially correct. The annotation was carried out in this manner as the main goal of the study was to learn a strategy for identifying clues that could assist in steering a receiver to a correct guess and certainly clues capable of eliciting partially correct guesses could be used in a clue sequence with other effective clues to such an end. We might also consider a more strict correctness measure that assigns zero or partial credit to partially correct guesses. We might also similarly reward other kinds of guesses that can lead to an ultimately successful guess such as guesses composed of a synonym of the correct guess.
We would like to collect more guess data using the design of the crowd sourced experiment that we conducted to bolster our results. It would also be interesting to extract additional textual features from the clues to see if they contain information that has predictive value for a clue's effectiveness. Such features could include the ratio of content to function words in a clue and features related to the POS tags of the words that compose the clue. Regarding which metric to use for human judgements, it is clear that in many cases the upper bound is too optimistic. An analysis of the RDG-Phrase corpus clearly shows the synergistic effects of clue combinations clearly contribute to steering the receiver to a correct guess as seen in Table 8. Clue sequences such as these clearly show how clues can build off of each other and a clue sequence can lead to a correct guess even if most of the clues in the sequence might be of little value in isolation.

8. Conclusion

We have improved on earlier work on building an automatic clue-giving agent for a word-guessing game. First, we introduced a new automatically generated clue corpus (for a list of almost 1,000 common nouns) that contains close to 200,000 clues using methods discussed in (Pincus et al., 2014). Second, we present an auxiliary data-set of audio recordings (Pincus and Traum, 2016) from a crowd-sourced experiment conducted to create a gold standard for clue effectiveness where a clue is deemed effective if it was capable of eliciting a correct or partial correct guess from the experiment participant. Third, we have discussed results from a machine learning experiment that used automatically extracted textual features from the automatically generated clues in order to predict the effectiveness of an arbitrary clue that is significantly higher than a baseline method of random selection and in line with an expected guessability of human clue-giving ability. Finally, we have defined several scoring methodologies to rank human clue-giving ability.
Future work includes additional data collection of the type described here as well as exploration of additional features to improve the automatic classifier's effective clue precision. We also intend to design effective clue classification experiments that evaluate the average number of effective clues per target-word, the effectiveness of different clue sequences as well as how prior guess(es) can be used to steer

the receiver to a target-word faster. Finally, we plan to explore additional scoring methodologies for guesses and effective clues.

9. Acknowledgements

We would like to thank Chris Wienberg and Ramesh Manuvinakurike for help with this work.
The effort described here is partially supported by the U.S. Army. Any opinion, content or information presented does not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.
This material is also based upon work supported by the National Science Foundation under Grant No. IIS-1219253.

10. Bibliographical References

Burgener, R. (2006). Artificial neural network guessing method and game, October 12. US Patent App. 11/102,105.

Burton, K., Java, A., and Soboroff, I. (2009). The icwsm 2009 spinn3r dataset. In *Proceedings of the Third Annual Conference on Weblogs and Social Media (ICWSM 2009)*.

Dumais, S., Platt, J., Heckerman, D., and Sahami, M. (1998). Inductive learning algorithms and representation for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management. ACM*.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. In *SIGKDD Explorations*, volume 11.

Higashinaka, R., Dohsaka, K., and Isozaki, H. (2007). Learning to rank definitions to generate quizzes for interactive information presentation. In John A. Carroll, et al., editors, *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*. The Association for Computational Linguistics.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, Nov.

Paetzel, M., Racca, D. N., and DeVault, D. (2014). A multimodal corpus of rapid dialogue games. In *Language Resources and Evaluation Conference (LREC)*, May.

Pincus, E. and Traum, D. (2014). Towards a multimodal taxonomy of dialogue moves for word-guessing games. In *10th Workshop on Multimodal Corpora*, May.

Pincus, E., DeVault, D., and Traum, D. (2014). Mr. clue-a virtual agent that can play word-guessing games. In *Tenth Artificial Intelligence and Interactive Digital Entertainment Conference*.

Sawaki, M., Minami, Y., Higashinaka, R., Dohsaka, K., and Maeda, E. (2008). ?who is this? quiz dialogue system and users' evaluation. In *2008 IEEE Spoken Language Technology Workshop*.

11. Language Resource References

Eli Pincus and David Traum. (2016). *Gold Standard Clue Effectiveness Corpus*. University of Southern California Institute for Creative Technologies, 1.0.