Context-Dependent Error Correction of Spoken Referring Expressions

Ingrid Zukerman, Andisheh Partovi, Su Nam Kim
Clayton School of Information Technology, Monash University
Clayton, Victoria 3800, Australia
Ingrid.Zukerman@monash.edu, Andi.Partovi@monash.edu, Su.Kim@monash.edu

Abstract

We integrate a supervised machine learning mechanism for detecting erroneous words in the output of a speech recognizer with a two-tier error-correction approach that features (1) a noisy-channel model that replaces erroneous words with generic words, and (2) a phonetic-similarity mechanism that refines the generic words based on a short list of candidate interpretations. Our results, obtained on a corpus of 341 referring expressions, show that the first tier improves interpretation performance, and the second tier yields further improvements.

1. Introduction

In recent times, there have been significant improvements in automatic speech recognition (ASR) [1]. Nonetheless, ASR errors still impede the widespread adoption of Spoken Language Understanding (SLU) systems. In this paper, we offer a context-dependent approach to error correction of spoken referring expressions which integrates a supervised machine learning mechanism for detecting erroneous words in ASR output [2] with a two-tier error correction component. This component features a noisy-channel model that replaces erroneous words with generic words on the basis of the expectations of a semantic model [3], and a phonetic-similarity mechanism that refines the generic words based on a short list of candidate interpretations.

To illustrate the workings of our system, consider a text such as “the built played on the table” (misheard from the description “the blue plate on the table”) in the context of household assistance — the application domain chosen for our implementation [4]. The error-detection component first determines that “built” and “played” are wrong. Then the noisy-channel model replaces “built played” with “UNKNOWN THING” to correct the mismatch between the parts-of-speech (PoS) of the heard words and the PoS expected for the object of a description (e.g., adjectives and nouns). The rationale for this replacement is that “built played” will lead a parser astray, thus subverting the interpretation process, while the replacement allows an SLU system to produce a reasonable syntactic, and then semantic, interpretation of the description. This in turn supports the generation of pragmatic interpretations comprising things on tables. Although having these interpretations is significantly better than having no interpretations, there may be many things on tables, which minimally would result in a clarification sub-dialogue. In order to alleviate this problem, the phonetic-similarity component determines which of the things on tables could be referred to by a word that sounds like “played”. At present, this approach is applied only to names of things (and not to attributes), because objects may have many attributes and combinations thereof (e.g., colour, size, texture, composition). The extension to attributes is left for future work.

Note that our approach differs from that adopted by SLU systems that prevent ASR errors by constraining the vocabulary and grammatical constructs understood by an ASR [5, 6, 7]. These approaches can process expected utterances efficiently, but have trouble processing unexpected utterances. In contrast, we first obtain candidate interpretations of an utterance without constraining the vocabulary, and then use phonetic attributes of plausible candidate interpretations to improve our results.

The error-detection and noisy-channel components are implemented as pre-processing steps of the Scusi? SLU system [2, 3, 4], and the error-correction component constitutes a post-processing step which re-ranks the top N interpretations produced by Scusi?. Our mechanism was evaluated on a corpus of 341 spoken referring expressions, which contains 209 descriptions for which all the textual interpretations returned by our ASR (Microsoft Speech SDK 6.1) for each description were wrong. The noisy-channel model improved the interpretation performance of the original Scusi? system, and the phonetic-similarity approach yielded further improvements (Section 5).

The rest of this paper is organized as follows. In the next section, we discuss related work. In Section 3, we outline the design of our system, and describe our error-detection and noisy-channel models. Our phonetic-similarity error-correction mechanism appears in Section 4. In Section 5, we discuss our evaluation, and then present concluding remarks.

2. Related Work

This research combines three elements: ASR error detection, error correction using a noisy-channel model and shallow semantic parsing, and refinement using phonetic similarity.

The noisy-channel model has been used for ASR error detection and correction [8] and disfluency correction [9, 10] in SLU systems. Other approaches to error detection and correction include [11, 12]. Zhou et al. [11] perform error detection in Mandarin using the Generalized Word Posterior Probability of the words in an utterance and features based on N-best hypotheses, and error correction using a model based on mutual information and trigrams. López-Cózar and Griol [12] use lexical approaches to replace, insert or delete words in an ASR output, and syntactic approaches to modify tenses of verbs and grammatical numbers to better match grammatical expectations.

Shallow semantic parsers for SLU systems have been used in [13, 14]. Coppola et al. [13] use FrameNet [15] to detect and filter the frames for target words, and employ a Support Vector Machine classifier [16] to perform semantic labeling. Geertzen [14] uses a shallow parser to detect semantic units only when a dependency parser fails to produce a parse tree.

Like these researchers, we offer corpus-based techniques to detect ASR errors. However, we employ features of the ASR output to build a classifier for error detection, rather than actual

1 Although there are better ASRs, the output of the Microsoft ASR constitutes an appropriate vehicle for research on error correction.
words or expectations from the linguistic context. This classifier, combined with a model that encodes semantic expectations, is used for noisy-channel error correction, whose results are refined using context-dependent phonetic-similarity.

3. System Design

Scusi? is a system that implements an anytime, probabilistic mechanism for the interpretation of spoken utterances, focusing on a household context. It has four processing stages, where (intermediate) interpretations in each stage can have multiple parents in the previous stage and can produce multiple children in the next stage, early processing stages may be probabilistically revisited, and only the most promising options at each stage are explored further.

Scusi? workflow. The system takes as input a speech signal, and uses an ASR (Microsoft Speech SDK 6.1) to produce candidate texts. Each text is assigned a score given the speech wave. Next, Scusi? applies Charniak’s probabilistic parser (http://www.ai.sri.com/projects/likelihood) to syntactically analyze the texts, yielding at most 50 different parse trees per text. The third stage applies mapping rules to the parse trees to generate Uninstantiated Concept Graphs (UCGs) [17] that represent the semantics of the utterance. The final stage instantiates the UCGs with objects and relations from the current context. For example, given a parse tree for “the blue plate on the table”, the third stage returns the UCG plate\text{COLOR}: blue on\text{–on}: table (different parse trees yield different UCGs). The final stage returns candidate Instantiated Concept Graphs (ICGs), e.g., plate 1 – location on\text{–on}: table 2, plate 2 – location on\text{–on}: table 1, ranked in descending order of probability.

The probability of each ICG depends on (1) how well the concepts and relations in it match the corresponding concepts and relations in its parent UCGs, e.g., whether plate 1 can be called “plate”, and whether it is blue (recall that an ICG can have more than one parent UCG); and (2) how well the relations in the ICG match the relations in the context, e.g., whether plate 1 is indeed on\text{–on}: table 2. The details of the calculation of the probability are described in [4]. The aspect that is most relevant to this paper is the estimation of the probability that a word heard by the ASR, \( h \), refers to the corresponding instantiated concept \( k \) in an ICG, e.g., the probability that “saucer” refers to plate 1. Scusi? estimates this probability using Leacock and Chodorow’s Wordnet similarity metric [18] between \( h \) and a canonical word used to designate \( k \) (e.g., the canonical word for plate 1 is “plate”). The similarity scores are normalized to the [0, 1] range, yielding \( \text{Pt}_k(h,k) \), the probability that word \( h \) refers to \( k \) (0 probabilities are smoothed to an arbitrarily low value \( c \) in order not to invalidate any interpretation).

Error detection and correction. The error-detection process receives as input alternative texts produced by the ASR, and activates a classifier that determines whether a word in a text is correct [2]. The error-correction process is divided into pre- and post-processing stages. In the pre-processing stage, we employ a shallow semantic parser (SSP) that assigns semantic labels to segments in the text, and a noisy-channel error corrector that decides on alterations to be made to the ASR output based on the expectations of a semantic model [3]. The resultant (possibly modified) texts are given as input to Scusi?. The top \( N \) interpretations (ICGs) generated by Scusi? are then passed to the post-processing stage, where the phonetic-similarity module re-ranks these ICGs based on the phonetic similarity between erroneous words in the head noun positions and words commonly used to refer to the corresponding instantiated concepts in these ICGs (e.g., plate 1 and table 2).

To illustrate this process, let us revisit the example in Section 1: “the built played on the table”. The PoS tags for the italicized words, viz VBN and VBD, do not match the expectations of Scusi?’s semantic model (Section 3.3). Hence, the pre-processing error-correction stage generates “the \text{UNKNOWN THING} on the table”, for which Scusi? produces candidate interpretations that comprise things on tables in our current context, e.g., mug 1–location on\text{–on}: table 2, plate 1–location on\text{–on}: table 1, mug 2–location on\text{–on}: table 1, etc. These interpretations are equiprobable, as there is no information to distinguish between them. The phonetic-similarity module now considers the objects that were instantiated to match \text{THING} (i.e., mug, plate, mug 2, etc), and phonetically matches the heard word “played” with words that could be used to refer to these objects, which are obtained from WordNet [19] (e.g., “plate”, “dish”, “saucer” designate plate 1). In this case, “plate” has a better match with “played” than the other options, which results in the interpretations comprising plates, including plate 1, being ranked ahead of the other interpretations.

We now outline the word-error classifier and SSP, and report on their performance, followed by a brief description of the noisy-channel model. The phonetic-similarity module is described in Section 4, and evaluated in Section 5, together with the noisy-channel model. The performance of the classifier and SSP was evaluated over the complete corpus constructed to evaluate the Scusi? system, which comprises 400 spoken descriptions generated by 26 speakers [4]. We performed 13-fold cross-validation, where each fold contains two speakers.

3.1. Word Error Classifier

We investigated Decision Trees [20] and Naïve Bayes classifiers [21] (cs.waikato.ac.nz/ml/weka/). The best performance was obtained by the Decision Trees, which yielded an average accuracy of 92% over the 13 cross-validation folds. The most influential features are: PoS; Broad Sound Groups (BSGs) – a vector of length 8 that represents the number of times each BSG occurs in a word; and Phonemes – a vector of length 39 that represents the number of times a phonetic symbol appears in a word. These features were calculated for the current, previous and next word.

3.2. Shallow Semantic Parser (SSP)

We found the following semantic labels useful for descriptions:
- **Object** – a noun group designating an object in the world, e.g., “the blue ceramic drinking mug on the table”.
- **Preposition** – a preposition or prepositional expression, e.g., “on” or “further away from”.
- **Landmark** – like Object, but a description may have several Landmarks, e.g., “the mug on the table in the corner”.
- **Noise** – sighs or hesitations that are often mis-heard by the ASR as “and”, “on” or “in”.
- **Specifier** – a further specification that normally precedes a Landmark, e.g., “the center of”, “front of” or “the left of”.

We employed the Mallet implementation of the linear chain Conditional Random Fields (CRF) algorithm [22] to learn sequences of semantic labels (mallet.cs.umass.edu). CRF was trained and tested on 800 annotated samples from the ASR output: 400 from the best output and 400 from the worst. The features (for each word) that yielded the best performance are: PoS; relative position; and whether the word is “of”, a stop word, a preposition, or the designation of a location (e.g., “center”, “left”). SSP obtained an average F-score of 0.85 over the 13 cross-validation folds. However, its performance for Noise was rather poor, with an average F-score of 0.69 (precision was 0.91, but recall was 0.56).
3.3. Noisy-channel Model

Given a textual output produced by the ASR, our noisy-channel model removes Noise, inserts missing prepositions and replaces erroneous words. The decision to perform removals and replacements depends on the probability assigned by the classifier to the words in question being wrong, and the impact of the action on the probability of the resultant word sequence according to the semantic model [3]. Specifically, words whose PoS differs from that expected by the semantic model for a particular semantic segment (obtained from textual transcriptions of spoken descriptions) are likely to be replaced. For example, according to our semantic model, Objects and Landmarks expect D/NN for the middle words, and NN for the last word. If the words assigned to an Object or Landmark do not meet these expectations, they may be replaced by generic words such as “the”.

4. Phonetic similarity module

This module receives as input a ranked list of N ICGs. As mentioned above, the probability of an ICG incorporates the probability of the lexical match between instantiated concepts in Objects and Landmarks and the corresponding head nouns in parent UCGs. This head noun could be the word THING (which replaces non-nouns) or a noun heard by the ASR. Figure 1 illustrates this situation for the spoken description “the plate on the table”, where the ASR has produced three textural interpretations: “the plane on the table” (T1) and “they played/implied on the table” (T2, T3). The noisy channel model replaces “they” with “the” and the verbs with THING, merging T2 and T3 into the textural interpretation “the THING on the table” (“plane” is a noun, and hence it is not changed). Each text eventually yields a UCG (U1 and U2), from which ICGs I1, I2 and I3 are generated using objects and relations from the context [4]. These ICGs are equiprobable, as the two mugs and the plane have similar lexical similarity probabilities with THING (the probabilities are not the same due to the workings of the Leacock and Chodorow metric [18]), and a similarity probability of ε with “plane”.

4. Phonetic similarity module

This module receives as input a ranked list of N ICGs. As mentioned above, the probability of an ICG incorporates the probability of the lexical match between instantiated concepts in Objects and Landmarks and the corresponding head nouns in parent UCGs. This head noun could be the word THING (which replaces non-nouns) or a noun heard by the ASR. Figure 1 illustrates this situation for the spoken description “the plate on the table”, where the ASR has produced three textural interpretations: “the plane on the table” (T1) and “they played/implied on the table” (T2, T3). The noisy channel model replaces “they” with “the” and the verbs with THING, merging T2 and T3 into the textural interpretation “the THING on the table” (“plane” is a noun, and hence it is not changed). Each text eventually yields a UCG (U1 and U2), from which ICGs I1, I2 and I3 are generated using objects and relations from the context [4]. These ICGs are equiprobable, as the two mugs and the plate have similar lexical similarity probabilities with THING (the probabilities are not the same due to the workings of the Leacock and Chodorow metric [18]), and a similarity probability of ε with “plane”.

Now that we have plausible candidate ICGs, i.e., objects on tables (instead of all the objects in a room), the phonetic-similarity module distinguishes between these ICGs based on their phonetic similarity with the heard words. This is done by applying the following procedure to each instantiated concept k in each of the top-N ICGs, where N = 10 in our current trials (there is one instantiated concept in the Object and one in each Landmark of an ICG).

1. Inspect each of the ICG’s parent UCGs to find words of interest linked to k. These words may be either THING (used to replace one or more non-nouns) or heard nouns that have a lexical probability of ε of referring to k. For instance, in Figure 1, THING is obtained from non-nouns “played” and “implied”, and the heard noun corresponds to “plane”.

2. Compute the phonetic edit distance between a heard word hw i obtained in the previous step and each of k’s synonyms, syn j(k) for j = 1, 2, ..., 2. The similarity between two phonemes depends on their BSG: phonemes belonging to the same BSG are considered more similar to each other than to phonemes belonging to different BSGs. In addition, we have grouped similar sounding vowels (e.g., iy, ih, ax belong to the same group), and set their similarity scores so that vowels in the same phonetic group are considered more similar to each other than to vowels in different groups. The resultant edit distance is mapped to the [0,1] range to yield Pr phon (hw i, syn j (k)), the probability of the ASR hearing hw i given that the speaker uttered syn j(k).

3. Compute the probability that the ASR heard hw i given that the intended object is k:

$$Pr_{joint}(hw_i|k) = \max_j [Pr_{phon}(hw_i, \text{syn}_j(k)) \times Pr_{lex}(\text{syn}_j(k)|k)]$$

This probability jointly takes into account the phonetic similarity between a heard word hw i and each synonym of k, and the lexical similarity between this synonym and k’s canonical word.

4. Use Pr joint (hw i|k) to recalculate the probability of the ICG, where Pr joint (hw i|k) replaces Pr lex (hw i|k) for each of the heard words identified in Step 1 (viz “played”, “implied”, “plane”). We consider two replacement policies: P THING and P ALL. P THING replaces Pr lex (hw i|k) only in ICGs where THING refers to k in at least one textual interpretation (i.e., at least one text contains a non-noun, which is indicative of a particularly bad ASR output). The replacement is performed only for heard non-nouns that led to THING, e.g., “played” and “implied” in Figure 1. In addition to the replacements made by P THING, P ALL replaces Pr lex (hw i|k) in ICGs where all the heard nouns corresponding to k have an ε probability of referring to k, e.g., “plane” in Figure 1 (if there is a heard noun with a non-ε probability of referring to k, nothing is done).

Note that P THING requires the consideration of the relative contribution of each of the non-nouns that led to THING to the probability of an ICG. For instance, say “played” appeared in two textual interpretations and “implied” in one, prior to being merged into one text that contains THING. The contribution of these non-nouns is taken into account by splitting the UCG containing THING into several UCGs, one for each heard non-noun. The resultant UCGs differ only in their contribution to the probability of their child ICG. For example, “plate” is more similar to “played” than to “implied”, hence Pr joint (“played”|k) > Pr joint (“implied”|k).

This process ranks the ICGs whose instantiated concepts sound like the heard words ahead of other ICGs, e.g., plate=location_on=table in Figure 1 is ranked first in our example.
5.1. Corpus

Performance was evaluated using the 341-description corpus with which we evaluated the original Scusi? system [4] (obtained by removing 59 descriptions that cannot be represented by Scusi? from the original 400-description corpus). The descriptions in the corpus, which vary in length and complexity, have an average description length of 10 words. Sample descriptions are: “the green plate next to the screwdriver at the top of the table”, “the large pink ball in the middle of the room”, “the plate on the corner of the table” and “the computer under the table”. As mentioned in Section 1, 209 of these descriptions do not have a correct textual interpretation.

5.2. Evaluation metric

We employed the evaluation metric Normalized Discounted Cumulative Gain (NDCG) @K [23], which takes into account equiprobable interpretations by allowing the definition of a ‘relevance measure’ for a result; and provides a finer-grained account of rank than the commonly used Recall measure by dividing the relevance measure by a logarithmic penalty that reflects the rank of the result. Using $f_c(I_j)$ as a measure of the relevance of interpretation $I_j$, we obtain

$$DCG@K(d) = f_c(I_1) + \sum_{j=2}^{K} \frac{f_c(I_j)}{\log_2 j},$$

where $f_c$ is the fraction of correct interpretations among those with the same probability as $I_j$ (this is a proxy for the probability that $I_j$ is correct):

$$f_c(I_j) = \frac{c_j}{h_j - l_j + 1},$$

where $l_j$ is the lowest rank of all the interpretations with the same probability as $I_j$, $h_j$ is the highest rank, and $c_j$ is the number of correct interpretations between rank $l_j$ and $h_j$, inclusively.

$DCG@K$ is normalized to the $[0, 1]$ range by dividing it by the score of an ideal answer where the interpretations in the set of correct interpretations for description $d$, $|C(d)|$, are ranked in the first $|C(d)|$ places, yielding

$$NDCG@K(d) = \frac{DCG@K(d)}{1 + \sum_{j=2}^{\min(|C(d)|, K)} \frac{1}{\log_2 j}}.$$

5.3. Results

To analyze our results, we separated the 209 descriptions for which the ASR did not produce a correct textual interpretation into 131 descriptions that led to THING as a referent of an instantiated concept $k$ (and can potentially be corrected by $P_{THING}$), and 78 descriptions that led only to nouns (and can potentially be corrected only by $P_{ALL}$). As mentioned in Section 4, the first category is deemed more garbled than the second.

Analysis of these two categories reveals that, as expected, for the first category, Scusi?+PS (with $P_{THING}$) outperforms Scusi? (statistically significant at $p$-value $< 0.01$ for all $K$), and Scusi?+NCM outperforms Scusi? (statistically significant at $p$-value $< 0.01$ for $K \geq 3$). Scusi?+PS also outperforms Scusi?+NCM (not significant). However, for the second category, the original Scusi? system outperforms Scusi?+PS (with $P_{ALL}$), which in turn outperforms Scusi?+NCM (not significant). In addition, the original Scusi? system outperforms both error-correction policies, as well as the noisy-channel model, with respect to the 132 descriptions for which one of the texts generated by the ASR is correct.

These observations prompt us to propose a hybrid system, where Scusi?+PS (with $P_{THING}$) is used for any description that leads to THING in the pre-processing stage, and the original Scusi? system is used for all other descriptions. We also tested a hybrid version of Scusi?+NCM for comparison purposes.

Figure 2(a) depicts the performance of the different versions of Scusi? with respect to the 209 descriptions for which the ASR produced only incorrect textual interpretations. All the enhanced versions of Scusi? outperform the original Scusi? system (statistically significant at $p$-value $< 0.01$ for all $K$ for Scusi?Hybrid+$P_{THING}$, and at $p$-value $< 0.05$ for $K \geq 3$ for the other versions). Scusi?Hybrid+$P_{THING}$ also outperforms the other versions (statistically significant at $p$-value $< 0.05$ compared to Scusi?+NCM for $K = 1$). These results are consistent with the results obtained for the entire corpus, depicted in Figure 2(b). Importantly, the largest improvements are obtained for interpretations with low (good) ranks ($K = 3$) — recall that the phonetic-similarity procedure re-ranks only the top $N$ ($=10$) ICGs, hence changes in the NDCG of higher ranks take place due to flow-on effects.

6. Conclusion

We have offered an error-correction approach for spoken descriptions which integrates a supervised machine learning mechanism for detecting erroneous words in ASR output with a two-tier error correction component that features (1) a noisy-channel model that replaces erroneous words with generic words, and (2) a phonetic-similarity mechanism that refines the generic words based on a short list of candidate interpretations. Our results show that this approach yields significant improvements in interpretation performance, in particular with respect to interpretations with low (good) ranks, which form the basis for further interactions with users.
7. References


