A Probabilistic Approach to Analogical Speech Translation

Keiko Horiguchi & Alexander Franz
D21 Laboratory, Sony Corporation
6-7-35, Kita-Shinagawa, Shinagawa-ku, Tokyo 141, Japan
Email: {keiko,amf}@pdp.crl.sony.co.jp

Abstract

Previous work on speech-to-speech translation has suffered from problems of brittleness and low quality (rule-based approaches), or from excessive data requirements and linguistic inefficiency (analogical or example-based approaches). In this paper, we present a probabilistic approach to analogical speech translation, and describe its integration with linguistic processing. The evaluation results show that this approach results in high-accuracy translations in limited domains.

1 Introduction

Recent advances in speech recognition technology have enabled research on robust, accurate speech-to-speech translation. This entails a number of formidable challenges.

1.1 Problems of Spoken Language

Spoken language has many properties that do not appear in written language. One class of such properties are speech performance errors, meaningless (for our purposes) by-products of the speech production process. Speech performance errors include errors in pronunciation, word selection, and structure selection. If an error is corrected by the speaker, this results in a repair or restart following an interruption of the utterance, possibly including a word fragment. In addition to these problems, a speech translation module also has to handle the errors introduced by the speech recognition component.

1.2 Natural Speech Properties

In contrast to e.g. natural language interface systems, such as (Jackson et al., 1991), where only the propositional content of the speaker's query has to be extracted and mapped onto an unambiguous system command, a speech translation system needs to process much more information from the input utterance.

When a speech translation system is used to translate a human-to-human verbal exchange, "interpersonal meaning" has to play a large role in the translation. Many phenomena that are produced intentionally by the speaker and that carry specific pragmatic, communicative functions deviate far from standard written grammar.

For example, interjections and filled pauses or hedges often appear in the middle of utterances, expressing the speaker's hesitation or calling attention to the immediately following words or phrases. Incomplete sentences are often used to soften speech acts that might have negative effects on the listener (such as a rejection or an imposing request). For these reasons, techniques that can be used in a natural language interface for ignoring these phenomena and extracting the propositional content can not be applied in speech translation of human dialogues, lest the output become dull, mechanical, and pragmatically inappropriate.

1.3 Rule-based Speech Translation

Traditional rule-based spoken language systems address these problems by trying to process spoken input with a written-language grammar, and then attempting to recover from analysis failures (Seneff, 1992). On the whole, rule-based speech translation faces problems of brittleness, being ill-suited to the characteristics of spoken language, and less than perfect output quality.

2 Analogical Speech Translation

Analogical (or example-based) translation (Nagao, 1984) relies on a database of pre-translated bilingual example pairs. The source language input expression is matched against the source language portion of each example pair, and the best matching example is selected. The system then returns the target language portion of the best example as output. Pre-translation results in high quality, and the matching process can be very robust, yet (if appropriate examples are present) make very fine distinctions.

Unfortunately, the pure analogical approach lacks scalability. The effort required to acquire the examples, the cost of the space required to store the examples, and the cost of the time required to search the database can become prohibitively high, since a pure analogical system requires a separate example for every linguistic variation.

3 Probabilistic Analogy

Viewing translation by analogy within a probabilistic framework yields what we have called the "cascaded

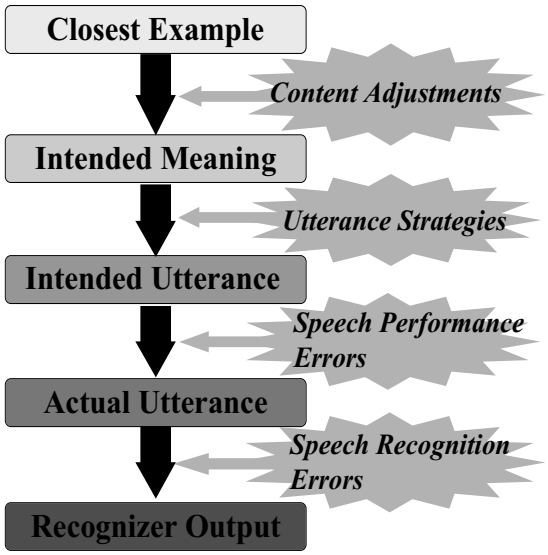

Figure 1: Cascaded Noisy Channel Model

noisy channel model" (Figure 1). In this model, the speaker first selects an example E that is closest to the message that she intends to express. Then, the speaker adjusts the contents by replacing words or phrases, and by adding or deleting modifiers, such as adverbial phrases and adjuncts. This process yields the intended meaning. Next, depending on the speech situation and context, the speaker applies certain pragmatic utterance strategies, which results in emphasizing or omitting certain parts of the contents. This yields the intended utterance, which is characterized by natural speech properties such as ellipsis, inverted word order, or interjections.

When the speaker actually vocalizes the utterance, speech performance errors such as mispronunciations, grammatical errors, or restarts occur. The result of this is the actual utterance that is presented to the listener. The first step of the speech translation system is the speech recognition component, which introduces recognition errors. The result of this is the recognizer output.

Thus, the recognizer output, which represents the input to the translation engine, has traversed four distinct channels or distortion processes, each of which is associated with different causes and effects on the message. Previous research has shown that speech recognizer errors can be modeled, and corrected, in the noisy channel framework (Ringger and Allen, 1996). In our work, we extend this approach to cover a sequence of separate sources of distortions.

4 Deriving the Probabilistic Model

This section describes the details of the probabilistic model. Let I denote the input expression, consisting of a sequence of words along with certain features resulting from linguistic analysis. Thus, I consists of a sequence of words $\mathrm{iw}_1, iw_2, \ldots, iw_n$, and a set of features $\mathrm{if}_1, \mathrm{if}_2, \ldots, \mathrm{if}_m$. Similarly, let the source expression E of an example pair consist of $\mathrm{ew}_1, \mathrm{ew}_2, \ldots, \mathrm{ew}_p$ and $\mathrm{ef}_1, \mathrm{ef}_2, \ldots, \mathrm{ef}_q$.

Given an input expression, an analogical translation algorithm must determine the example pair that is closest in meaning to the input expression. We denote the probability that an example expression is appropriate for translating some input as the conditional probability of the example, given the input:

(1) $P(\text{Example}|\text{Input})$

Our aim is to find the example $E_{\max}$ that has the highest conditional probability of being appropriate to translate the given input, where the **max** function chooses the example with the maximum conditional probability:

(2) $E_{\max} = \mathbf{max}_{E \in \text{Examples}} [P(E|I)]$

Applying Bayes' law and ignoring $P(I)$ yields the following:

(3) $E_{\max} = \mathbf{max}_{E \in \text{Examples}} [P(E)P(I|E)]$

The probability distribution over the examples $P(E)$ encodes the prior probability of using the different examples to translate expressions in the domain. The conditional probability distribution $P(I|E)$ is modeled using a number "distortion" operators for echoing, deleting, adding, and altering a word or a syntactic or semantic feature.

- **echo-word**(ew_i). This operator simply echoes the i^{th} word, ew_i, from the example to the input.
- **delete-word**(ew_i). This operator deletes the i^{th} word, ew_i, from the example.
- **add-word**(iw_j). This operator adds the j^{th} word, (iw_j), to the input.
- **alter-word**($\mathrm{ew}_i, \mathrm{iw}_j$). This operator alters the i^{th} word, ew_i, from the example to the j^{th} word, iw_j, in the input expression. The altered word is different, but usually semantically somewhat similar.
- Corresponding operators for features.

Given these operators, we can view the input I as an example E to which a number of distortion operators have been applied. Thus, we can represent an input expression I as an example E plus a set of distortion operators:

(4) $I = \{\mathbf{distort}_1, \ldots, \mathbf{distort}_x, E\}$

This means that we can re-express the conditional probability distribution for an input expression I, given that the meaning expressed by example E is intended, as follows:

(5) $P(I|E) = P(\{\mathbf{distort}_1, \ldots, \mathbf{distort}_x\}|E)$

A number of independence assumptions are required to make this model computationally feasible. First, we assume that the individual distortion operators are conditionally independent, given the example E.

Second, we make the assumption that the individual distortion operators only depend on the words and features that they directly involve. For example,

we assume that the probability of echoing a word depends only on the word itself. Similarly, we assume that the probability of e.g. deleting a feature depends only on the feature itself. This yields the following approximation:

$$(6) \quad P(I|E) \approx \prod_{k=1}^{x} P(\textbf{distort-word}_k(\mathrm{ew}_i, \mathrm{iw}_j)) \prod_{l=1}^{y} P(\textbf{distort-feature}_l(\mathrm{ef}_i, \mathrm{if}_j))$$

5 Match Computation

Given an input I and an expression E, it is straightforward to determine the probability of the feature distortion:

$$(7) \quad \prod_{l=1}^{y} P(\textbf{distort-feature}_l(\mathrm{ef}_i, \mathrm{if}_j))$$

Determining the probability of the word distortion requires us to find the most probable set of distortion operators:

$$(8) \quad \mathrm{Distort}_{\max} = \textbf{max}_{Distort}\,[P(\mathrm{Distort}|E, I)]$$

We accomplish this with a dynamic programming algorithm that finds a set of distortion operators with maximal probability. First, to obtain a distance measure, we take the negative logarithm of this expression:

$$(9) \quad -logP(\mathrm{Distort}|E, I)$$

Given that we have assumed independence between individual distortion operators above, this can be simplified as follows:

$$(10) \quad -log \prod_{k=1}^{\text{no. of operators}} P(\textbf{distort}_k|E, I)$$

We have also assumed that the distortion operators are independent of the part of the sentence that does not directly involve them. Thus, we can simplify further as follows:

$$(11) \quad -log \prod_{k=1}^{x} P(\textbf{distort}_k|\mathrm{ew}_i, \mathrm{iw}_j)$$

This can be further split into the individual distortion operators:

$$(12) \quad \sum_{k=1}^{x} -logP(\textbf{distort}_k(\mathrm{ew}_i, \mathrm{iw}_j)$$

This corresponds directly to with the individual costs that we use for the dynamic programming equation. Let the example expression be $E = e_1, e_2, \ldots, e_p$ and and the input expression be $I = i_1, i_2, \ldots, i_n$. Then, let $D(p, n)$ be the distance between the example and the input. This distance is defined by the following recurrence:

$$D(p,n) = min \begin{cases} D(p-1,n-1) - logP(\textbf{echo}(\mathrm{ew}_p)) \\ D(p,n-1) \quad - logP(\textbf{add}(\mathrm{iw}_n)) \\ D(p-1,n) \quad - logP(\textbf{delete}(\mathrm{ew}_p)) \\ D(p-1,n-1) - logP(\textbf{alter}(\mathrm{ew}_p, \mathrm{iw}_n)) \end{cases}$$

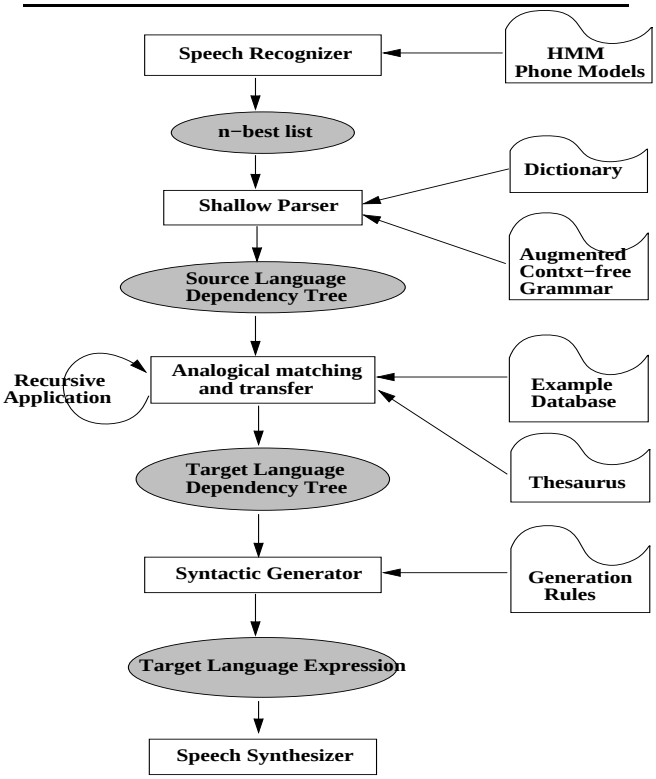

Figure 2: Overview of System Architecture

The result is the optimal alignment between the input and the example, as well as the minimum distance. The analogical matcher then selects the example with the smallest distance to the input.

6 Translation System Architecture

Based on this model, we have implemented a prototype speech-to-speech translation system that combines the flexibility and robustness of analogical translation with the linguistic efficiency and generality of linguistic rules.

The pipelined system architecture, shown in Figure 2, separates speech recognition, shallow parsing, and recursive analogical translation into different modules.

6.1 Morphological Analysis

Morphological analysis performs Part-of-Speech disambiguation, and dictionary and thesaurus look-up. In our prototype implementation, these operations are carried out by an adapted version of the JUMAN 3.1 Japanese morphological analyzer (Kurohashi et al., 1994).

6.2 Shallow Parsing

The next step performs parsing to a shallow degree, but with very high accuracy. The input is divided into clauses, noun phrases, and modifiers, and a shallow dependency tree is created. This function is currently performed with an augmented context-free

grammar for the NLYacc GLR parser (Ishii et al., 1994).

6.3 Recursive Analogical Translation

Probabilistic translation by analogy is applied recursively at the various linguistic levels to obtain a translation for the entire input expression. This step creates a shallow dependency tree in the target language.

6.4 Target Language Generation

The target language generation module performs a number of necessary linguistic operations, such as enforcing subject-verb agreement, and ensuring that required definiteness information is present (such as English determiners, quantifiers, or possessives). Then, the shallow dependency tree is linearized to create an expression or a sentence in the target language.

6.5 Speech Output

In the final step, spoken output is generated from the target language expression. In our Japanese-English prototype, this step is carried out by the DecTalk system (Hallahan, 1996).

7 Evaluation

We have implemented a prototype of this system to translate spoken Japanese input into English. The dictionary lists word classes and semantic categories, and it currently includes around 700 entries. The example database contains approximately 350 clause-level Japanese-English example pairs, and 700 phrase-level and word-level example pairs.

The evaluation was performed on an unseen test set of 150 expressions. Out of the total 150 expressions, we found that 2 expressions were impossible to translate well without more context, and thus beyond the scope of this type of system. From the remaining 148 sentences, 93% yielded good translations. Out of these good translations, 68% were without a flaw; 21% were missing grammatical markers; and 10% were marked by either a missing or an extra modifier.

8 Conclusions

Overall, the results of our evaluation showed that the system is able to achieve high-quality translation in in a limited domain. The probabilistic analogical translation step is able to overcome errors and "extra-grammatical" features in spoken language input, such as particle omissions, ellipsis, and metonymy. At the same time, the linguistic components introduce generality and linguistic efficiency that is essential for practical, speech-to-speech translation.

In future work, we are planning to address the problems identified in the evaluation by adding a linguistic processing step to extract predicate-argument structure. In addition, we are refining the shallow dependency tree matching algorithm to account for differences in modifier patterns, to perform better slot matching, and to improve the recursive translation mechanism.

For the longer term, we are considering ways to improve the integration between speech and spoken language processing, using more robust shallow analysis methods based on lexical statistics, and extending the system to cover additional languages.

References

Brown, P., Pietra, S. A. D., Pietra, V. J. D., and Mercer, R. L. (1993). The mathematics of statistical machine transalation: Parameter estimation. *Computational Linguistics*, 19(2):263–312.

Hallahan, W. (1996). DECtalk software: Text-to-speech technology and implementation. *Digital Technical Journal*, 7(4).

Ishii, M., Ohta, K., and Saito, H. (1994). An efficient parser generator for natural language. In *COLING-94*, pages 417–420, Kyoto, Japan.

Jackson, E., Appelt, D., Bear, J., Moore, R., and Podlozny, A. (1991). A template matcher for robust nl interpretation. In *Proceedings of the Speech and Natural Language Workshop*, pages 190–194.

Kurohashi, S., Nakamura, T., Matsumoto, Y., and Nagao, M. (1994). Improvements of Japanese morphological analyzer JUMAN. In *Proceedings of the International Workshop on Sharable Natural Language Resources*, pages 417–420, Nara, Japan.

Mayfield, L., Gavalda, M., Ward, W., and Waibel, A. (1995). Concept-based speech translation. In *ICASSP-95*, pages 97–100, Detroit, MI.

Nagao, M. (1984). A framework of a Machine Translation between Japanese and English by analogy principle. In Elithorn, A. and Banerji, R., editors, *Artificial and Human Intelligence*, pages 173–180. North-Holland.

Ringger, E. K. and Allen, J. F. (1996). A fertility channel model for post-correction of continuous speech recognition. In *ICSLP-96*, pages 897–900, Philadelphia, PA.

Seneff, S. (1992). A relaxation method for understanding spontaneous speech utterances. In *Proceedings of the Speech and Natural Language Workshop*, pages 299–304, Harriman, NY.

Shirotsuka, O. and Murakami, K. (1994). An example-based approach to semantic information extraction from Japanese spontaneous speech. In *ICSLP-94*, pages 91–94, Yokohama, Japan.

Sobashima, Y., Furuse, O., Akamine, S., Kawai, J., and Iida, H. (1994). A bidirectional, transfer-driven machine translation system for spoken dialogues. In *COLING-94*, pages 64–68, Kyoto, Japan.

Stemberger, J. P. (1982). Syntactic errors in speech. *Journal of Psycholinguistic Rsearch*, 11(4):313–333.