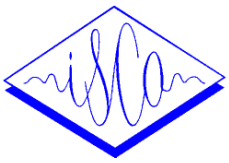

3rd European Conference on
Speech Communication and Technology
EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

SPOKEN LANGUAGE TRANSLATION WITH MID-90'S TECHNOLOGY: A CASE STUDY

Manny Rayner[1], *Ivan Bretan*[3], *David Carter*[1], *Michael Collins*[1], *Vassilios Digalakis*[2], *Björn Gambäck*[3], *Jaan Kaja*[4], *Jussi Karlgren*[3], *Bertil Lyberg*[4], *Steve Pulman*[1], *Patti Price*[2] *and Christer Samuelsson*[3]

(1) SRI International, 23 Millers Yard, Cambridge CB2 1RQ, UK
(2) SRI International, 333 Ravenswood Ave, Menlo Park, CA 94025, USA
(3) Swedish Institute for Computer Science, Box 1263, S-164 28 Kista, Stockholm, Sweden
(4) Telia Research AB, Rudsjöterassen 2, S-136 80 Haninge, Sweden

ABSTRACT

We describe[1] the architecture of the Spoken Language Translator (SLT), a prototype speech translation system which can translate queries from spoken English to spoken Swedish in the domain of air travel information systems. Though the performance given the level of effort so far has been extremely encouraging, more work is needed to provide a technology that will support widespread applications. With this goal, we have developed techniques for rapid development and for evaluation. These techniques allow us to estimate the level of effort required to achieve higher levels of performance.

10.21437/Eurospeech.1993-303

1. Introduction

How feasible is it to develop a high-quality spoken language translation system using technology that is available now, or can reasonably be expected to become available in the next two to five years? Based on our experiences in the SRI-SICS-Telia Research Spoken Language Translator (SLT) project, we argue that it is potentially feasible even in this relatively short time to build usable systems for limited domains and reasonably close language pairs, by adapting and configuring standard components in simple and well-defined ways. We will present our case by examining the performance of the current version of the system, which translates spoken English queries into Swedish in the domain of airline travel planning (ATIS), using a vocabulary of about 1000 words; we divide the errors made by the system into characteristic types, and consider the difficulties involved in attacking the problems that cause them to arise. To deal completely with some of these issues will undoubtedly require methods beyond the current state of the art. However, we will argue that the "difficult" problems occur fairly infrequently in observed corpus data, and that solving the simpler ones would be enough to produce a system to which users could adapt without undue strain.

The rest of the paper is organized as follows. In section 2, we briefly describe the SLT system: the main components, the interfaces between them, and the development methodology used. Section 3 examines the prototype SLT system's current level of performance, following which section 4 attempts to estimate how much the current performance could be improved, assuming that the present development strategy were consistently maintained over a period of two to five more years. Section 5 concludes.

[1]The research reported in this paper was sponsored by Swedish Telecom (Televerket Nät). Michael Collins is supported by a studentship from the UK Science and Engineering Research Council.

2. Overview of the SLT system

This section gives a brief overview of the SLT system; for a longer treatment, the reader is referred to [8]. At the highest level of generality, the guiding themes of the project has been those of *intelligent reuse of standard components* and *robust interfaces*. Most of the system is constructed from previously existing pieces of software, which have been adapted for use in the speech translation task with as few changes as possible.

We begin by describing the main system components, for speech recognition, text language processing, and speech synthesis. The speech recognizer used is a fast version of SRI's DECIPHER(TM) speaker-independent continuous speech recognition system [6]. It uses context-dependent phonetic-based hidden Markov models (HMMs) with discrete observation distributions for four features: cepstrum, delta-cepstrum, energy and delta-energy. The models are gender-independent and the system is trained on 19,000 sentences and has a 1381-word vocabulary. The output is an N-best hypothesis list, produced using a progressive recognition search [7].

Text processing for both languages is performed by the SRI Core Language Engine (CLE), a general natural-language processing system developed at SRI Cambridge [1]. Two copies of the CLE are used, equipped with English and Swedish grammars respectively. The English grammar is a large general-purpose feature grammar, which has been augmented with a small number of domain-specific rules. The Swedish grammar has been adapted fairly directly from the English one [4]. Each CLE grammar associates surface strings with representations in Quasi Logical Form (QLF; [2]), and can be compiled to run both in analysis mode (turning strings into QLFs), and generation mode (turning QLFs into strings). Generation is performed using a version of the Semantic Head-Driven generation algorithm [9].

Target language speech synthesis is performed by the

Swedish Telecom Prophon system [3], using polyphone synthesis. The polyphones are concatenated and the prosodic pattern determined by the Prophon analysis (which has access to the syntax and semantics of the utterance) is imposed via the PSOLA (pitch synchronous overlap add) signal processing technique [5].

The components are connected together in a pipelined sequence as follows. The input signal is processed by the recognizer, and a set of N-best hypotheses is passed to the English-language version of the CLE, each hypothesis tagged with an associated acoustic score. The CLE uses the grammar to analyze each speech hypothesis, and extract a set of possible QLF representations; this typically results in a set of between 5 and 50 QLFs per hypothesis. The *preference component* is then used to give each QLF a numerical score reflecting its *a priori* linguistic plausibility. The final score for the QLF is calculated as a weighted sum of the acoustic score for the speech hypothesis used and the preference score for the QLF, and the highest-scoring QLF is passed on to the next stage in processing. Since the preference component plays the crucial role in selecting the most plausible analysis hypothesis, we digress to describe its function in more detail.

The score produced by the preference component is a weighted sum of the scores returned by a set of about 30 individual *preference functions*. Preference functions are of two types. *Structural* preferences examine some aspect of the overall shape of the QLF. Typically, the number of occurrences of some relatively unlikely type of grammatical construction is counted, so that readings which contain instances of it can be penalized relative to those that do not. *Collocational* preferences, on the other hand, collect instances of $(Headword, Relation, Headword)$ triples in the utterance, where *Relation* is a grammatical relationship such as verb/subject, noun/preposition-phrase modifier or noun-noun combination. Each such triple then receives a score derived from its frequency of occurrence in some large corpus of previously analyzed domain utterances. The preference component is thus adapted to the domain in two ways. The collocational preferences reflect facts about the relative frequencies of co-occurrence of different head-words in the specified grammatical relations; also, the relative weights given to the individual preference functions can be tuned, using a statistical optimization method described in [8], to values appropriate for the given domain.

When the preference component has made its choice, the selected QLF is passed to the transfer component, which uses a set of simple non-deterministic recursive pattern-matching rules to rewrite it into a set of possible target-language QLFs [2]. A second application of the preference component is now performed, using a different set of preference functions, to select the most plausible transferred QLF. At the moment, the main work in the post-transfer preference component is performed by collocational preferences, which for example select between alternative translations of prepositions by rewarding common preposition/object pairs and penalizing uncommon ones. The highest-scoring QLF is then fed to the target-language grammar (running in generation mode), which converts it into a target-language surface string and an associated syntax tree. Finally, the string and tree are passed to the synthesizer.

In summary, the system has access to two main types of linguistic information. There is *rule information*, formulated as grammar rules, transfer rules and lexicon entries; and there is also *distributional information*, formulated explicitly as lists of frequencies of occurrence of certain types of local configuration, and implicitly as relative strengths to be assigned to the different preference functions. The rule information is applied to suggest possible hypotheses, which are then filtered through the distributional information to select the most plausible one. Rule information is more or less domain-independent: it captures the basic grammars of the source and target languages, and the cross-linguistic data defining the constructions in the target language that potentially can correspond to each construction in the source language. Distributional information is essentially domain-dependent: it captures the relative likelihood of the different rule-combinations permitted by the grammar (etc.) in the context of the given domain. This division of information sources dictates a development methodology in which three types of activity become central: development of domain-independent language descriptions, realized as sets of linguistically valid rules; collection of large domain corpora; and development of tools which can be used to process these corpora (semi-)automatically, to extract the distributional information inherent in them.

In the rest of the paper, we will discuss the extent to which this strategy has proven successful to date in the SLT project, and attempt to estimate the likely result that could be obtained by continued development along similar lines.

3. Current performance of the system

The modular nature of the SLT system's architecture suggests a methodology which evaluates the components separately. Our generic evaluation strategy will be to run a set of typical inputs through each component, and examine the output. For components that produce a non-deterministic set of outputs, success will be counted as producing at least one acceptable output; for preference filters, which receive a set of inputs and choose one "best" output, success will count as choosing an acceptable alternative from the possibilities available.

This simple picture should in fact be refined a little to take into account the fact that a speech translation system is by its nature interactive, which affects evaluation in at least two ways. Firstly, users can to some extent adapt to limitations of the system, if their nature is clear and if they do not pose insuperable obstacles to use. In particular, system performance in the current system degrades sharply

for sentences of more than about 12 or 13 words; however, a user who is aware of this can generally reformulate a long utterance as a sequence of two or more short ones, making the system's ability to deal with these examples correspondingly less important. Secondly, it may be possible to provide simple feedback to alert the user to possible processing errors, in effect adding an extra filter which aborts some sentences after failure to obtain user confirmation. The most obvious form of feedback is simply to echo back the selected speech hypothesis.

With these caveats, the system's performance from the user perspective is measured fairly well by three numbers. The first is the proportion of "short" utterances for which the system selects an acceptable speech hypothesis; the second is the proportion of correctly recognized sentences which receive a translation; and the third is the proportion of translations that are acceptable. Errors become progressively more serious as they occur further down the line. If the system fails to recognize the right hypothesis, this may be no more than an irritation: the user can try to correct by repeating or rewording the utterance. If the right hypothesis is heard, but no translation is produced, this is more serious: the user must assume that some linguistic construction is missing or incorrectly handled in the linguistic rules, and correction involves finding some suitable rephrasing of the utterance which hopefully will be within the system's coverage. The worst kind of errors are those in which an incorrect translation is produced, without the speaker being warned. Outputs can be incorrect in several ways. If the essential meaning is present in some form, the fact that output is spoken rather than typed can make grammatically ill-formed or stylistically disfluent output utterances hard to understand; though here, the target user can at least inform the source user that something has gone wrong, possibly permitting recovery. The absolute worst case is where the output utterance simply means something different from the input one, which may produce a situation where neither user is aware that communication has broken down.

The system's current performance figures, measured on previously unseen test data, are as follows. The tests were carried out on several hundred sentences, though not all at the same time; since the system is improving fairly rapidly, results have been rounded off to the nearest 5%. For sentences of length 12 words and under, 65% of all utterances are such that the speech hypothesis echoed back to the user is an acceptable one[2]. If the speech hypothesis is correct, then a translation is produced in 75% of the cases; and 90% of all translations produced are acceptable. Nearly all incorrect translations are incorrect due to their containing errors in grammar or naturalness of expression, with errors due to divergence in meaning between the source and target sentences accounting for less than 1% of all translations.

[2]The examples used were the utterances from the Fall 92 ATIS test set for which the reference sentence had 12 words or less (660 sentences), processed through an N-best interface with N=5. The figure 65% refers to the proportion of utterances for which the most preferred hypothesis within grammatical coverage was the same as or an acceptable variant of the reference sentence.

4. Projected future performance

Though the performance figures at the end of the above section are promising, they fall short of what would be needed for a genuinely useful system. The question now becomes that of estimating how much they could be improved; to answer this, we must examine more closely the sources of failure. We continue to consider only utterances of 12 words and less.

We begin by looking at the 35% of utterances for which no speech hypothesis, or an unacceptable speech hypothesis, is echoed back to the user. For a speech hypothesis to be chosen at all, at least one of the hypotheses in the N-best list passed from the recognizer has to be within the source language component's linguistic coverage; if more than one hypothesis is within coverage, the preference component has to select the right one. Of the 35%, about 22% fail due to no satisfactory hypothesis being in the N-best list; a further 7% because the correct hypothesis was outside coverage; and a further 6% because several hypotheses were within coverage, but the preference component chose the wrong one. Moving on to the cases where a correct speech hypothesis was found but no translation produced, the 25% of sentences that fail break down as about 14% due to missing target language or transfer rule coverage, about 7% due to missing source language coverage, and about 4% due to preference errors after source language analysis. The 10% of bad translations break down as about 5% due to missing transfer rule coverage, about 4% due to transfer preference errors (the post-transfer preference component is very new, and has hardly been tested yet), and 1% to source language analysis preference errors.

Now how much could this picture be improved in a five-year time-frame? Most obviously, we can expect some improvements in recognition quality as HMM-based technology continues to increase in maturity, and faster hardware platforms become available. It seems reasonable to hope that the 22% of utterances which fail at the recognition stage will shrink to 10% or less. What is more interesting to consider is the possible improvement that can be achieved by increasing linguistic coverage and refining the preference components.

Looking at errors due to missing linguistic coverage, we stress first that we are assuming development of the system's grammars and lexica as common resources that encode domain-independent linguistic information, and are shared among a number of projects: this has been the pattern to date in projects based on the Core Language Engine, and means that improvements made in one project automatically carry over to all the others. The crucial question is the extent to which coverage errors typically result from missing constructions *not* specific to the given domain; encouragingly, our findings here are that when coverage has

reached the current level of about 90%, most missing coverage is not in fact domain-specific. (Domain-specific items tend to occur with fairly high frequency, and will mostly already have been added when this point has been reached). The upshot is that we do not have to postulate five more years of work on producing increasingly refined coverage of a *single* domain, which is hard to justify on economic grounds; we can rather make a more reasonable projection on the basis of general coverage improvement in a number of domains.

Coverage improvement is a fairly mechanical process, which basically involves identifying holes and writing rules to fill them: the main question now is to estimate the distribution of the remaining holes, and how quickly they can be filled. A recent 200-sentence test run found 48 coverage failures (16 source language, and 32 transfer or target language). Somewhat arbitrarily, we will guess that any coverage problem that occurs at least twice in the current 5000-sentence training set will have been solved within five years; such problems are likely to recur in any reasonably-sized corpus and thus come to grammar developers' attention. On this basis, we could expect that all but three of the 48 failures would be solved within five years, indicating an increase in coverage to around 98-99%.

The central justification for our confidence in being able to increase linguistic coverage to a high level is that both the grammar and transfer rule formalisms are monotonic; addition of new rules can never invalidate old ones. The drawback is that new rules will in general produce new analysis and transfer hypotheses, which must be filtered out by the preference component. We assume that preferences will continue to be implemented on a statistical basis, since it seems entirely too optimistic to hope that robust reasoning tools will be available within a short-term perspective. It is of course not easy to predict what advances in statistical modelling will be practical; however, our experience is that if it is intuitively plausible that sufficient data exists in the training corpus to justify a preference decision on statistical grounds, then it is normally possible to construct a statistical preference function which realizes the intuition concretely.

Here it will be helpful to look at some of the preference failures from the test run mentioned above: thus for example in the sentence *"Is breakfast served on U A twenty one"*, the analysis incorrectly selected as best could be paraphrased as *"Is the breakfast which is served on U A the same as twenty one?"*. One piece of statistical corpus information which intuitively could be used to refute this interpretation is that the corpus contains several examples of sentences in which breakfast is referred to as being served, but none in which it is referred to as equal to anything (let alone a numbered object). Examining the eight examples of analysis preference errors, it is possible to construct similar arguments to deal with seven of them; the only case where no obvious solution is apparent is *"Which is your nonstop flight around eleven twelve or one o'clock?"*, where the preferred interpretation makes *"eleven twelve"* (i.e. twelve minutes past eleven) into a time expression. Preferring the correct interpretation here (i.e. the one where *"eleven twelve or one o'clock"* is a three-way disjunction) actually does seem to require some sort of non-trivial reasoning. If we are willing to accept this case study as reasonably typical, it gives us a tentative guess that preference coverage could also be increased to about the 98-99% mark by systematic extension of current methods.

5. Conclusions

Making fairly conservative extrapolations from the current SLT prototype, it seems to us reasonable to believe that simply continuing the basic development strategy could within five years produce an enhanced version, which recognized about 90% of the short sentences (12 words or less) in a specific domain, and produced acceptable translations for about 95-97% of the sentences correctly recognized. Since the greater part of the system's knowledge would reside in domain-independent grammars and lexicons, it would also be possible to port it to new domains with a fairly modest expenditure of effort, most of which would be concerned with collection of a sufficiently large domain corpus. It will be interesting to see whether these predictions come to pass; if they do, we feel it will provide an adequate demonstration that computational linguistics is mature enough to take another large step out of the research laboratories, and into the market-place.

References

1. Alshawi, H. (ed.), *The Core Language Engine*, MIT Press, 1992.
2. Alshawi, H., Carter, D., Rayner, M. and Gambäck, B., "Transfer through Quasi Logical Form", *Proc. 29th ACL*, Berkeley, 1991.
3. Ceder, K. and Lyberg, B., "Yet Another Rule Compiler for Text-to-Speech Conversion?", *Proc. ICSLP*, Banff, 1993.
4. Gambäck, B. and Rayner, M., "The Swedish Core Language Engine", *Proc. 3rd NOTEX*, Linköping, 1992.
5. Moulines, E. and Charpentier, F., "Pitch-Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones", *Speech Communication* Vol. 9, 1990.
6. Murveit, H., Butzberger, J. and Weintraub, M., "Speech Recognition in SRI's Resource Management and ATIS Systems", *Proc. DARPA Workshop on Speech and Natural Language*, 1991.
7. Murveit, H., Butzberger, J., Digalakis, V. and Weintraub, M., "Large Vocabulary Dictation using SRI's DECIPHER(TM) Speech Recognition System: Progressive Search Techniques", *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Minneapolis, Minnesota, April 1993, pp. II-319 - II-322.
8. Rayner, M., Alshawi, H., Bretan, I., Carter, D., Digalakis, V., Gambäck B., Kaja J., Karlgren J., Lyberg B., Price, P., Pulman, S. and Samuelsson, C., "A Speech to Speech Translation System Built From Standard Components". To appear in: *Proceedings of the ARPA workshop on Human Language Technology*, Plainsboro, NJ, 1993.
9. Shieber, S. M., van Noord, G., Pereira, F.C.N and Moore, R.C., "Semantic-Head-Driven Generation", *Computational Linguistics*, 16:30–43, 1990.