Commercial Applications of Natural Language Processing

Kenneth W. Church
AT&T Bell Laboratories
and
Lisa F. Rau*
Systems Research and Applications (SRA) Corporation
2000 15th Street North
Arlington, VA 22201

1 Introduction

Vast quantities of text are becoming available in electronic form, ranging from published documents (e.g., electronic dictionaries, encyclopedias, libraries and archives for information retrieval services), to private databases (e.g., marketing information, legal records, medical histories), to personal email and faxes. On-line information services are reaching the mainstream. There were over 15 million Internet users in 1993, and projections are for 30 million in 1997.

The world around us is experiencing revolutionary changes. With media attention reaching all-time highs, hardly a day goes by without a new article on the National Information Infrastructure, digital libraries, networked services, digital convergence or intelligent agents. These changes are conspiring to put natural language processing along the critical path for all kinds of novel applications.

This paper will mention a number of successful applications of natural language processing. Word processing and information management are two of the better examples, though there have been many others, both large and small. A small success, worth a few million or perhaps even a few tens of millions of dollars a year, is more than enough to support a small business. There have also been a few big successes, e.g., $100 million or more a year in revenues, large enough to help create whole new industries.

Of course, along with the successes, there have also been a few failures.

* *This paper was prepared while Lisa Rau was on an NSF Visiting Professorship for Women grant (NSF GER-9350134), hosted by the Computer and Information Sciences Department at the University of Pennsylvania.*

Figure 1: Continuum of NL Applications

We don't want to contribute to the "hype" by mentioning only the successes. The early boom and bust in machine translation (MT), starting with the first public demonstration of MT in 1954 and ending with the sobering findings of the ALPAC committee in 1966 [1], will be mentioned as an example of the danger of excessive optimism.

While keeping these lessons of the past in mind, this article will describe a few applications of natural language processing that have been profitable in the past, along with a few that appear promising for the future. Unfortunately, it is not possible to cover all of these successes in a single article. Speech recognition applications were recently surveyed in this magazine. Many other deserving applications such as OCR (optical character recognition) also had to be omitted.

2 Word Processing and Desktop Publishing

The commercial importance of word processing has been recognized for a long time. Back in 1966, the area, then known as computerized publishing, was cited in the ALPAC report (Appendix 17) as a good example of a commercially promising application area (in contrast with machine translation). As the ALPAC committee predicted, word processing has blossomed into a major industry. Microsoft Word and WordPerfect are regularly discussed in the financial pages of major newspapers around the world. Word processing is one of the better examples of a so-called "killer" application, the kind of application that can create a whole new industry. The lucrative word-processing market has become extremely competitive. If one vendor adds a natural language feature such as spelling correction, hyphenation or grammar/style checking, the others are sure to follow.

These features depend on technologies of varying degrees of difficulty. Some of these technologies, such as simple string matching, are so well understood that we feel uncomfortable referring to them as "natural language processing," while others such as grammar checking are so difficult that the technology may not be up to the task. WordPerfect (Novell) is offering Grammatik 6, an ambitious product that not only checks for grammatical errors, but even attempts to fix them. Microsoft has demonstrated a considerable long-term commitment to improving the technology by hiring a research group, many of whom are well known for significant contributions to grammar checking while at IBM [2].

What counts as natural language processing (NLP)? Simple string matching? Spelling correction? Grammar checking? DWIM (do what I mean [20])? HWIM (hear what I mean [21])? In artificial intelligence (AI), it has been said that once a problem becomes sufficiently well understood that it can be solved by a machine, it is no longer AI. Similar comments may apply here as well. If we know how to do it, then maybe it isn't NLP.

Figure 1 shows a number of technologies ranging from well-understood technologies, such as string matching, to more forward looking technologies such as grammar checking. At the left edge of the continuum, technologies such as table lookup are so well established that they are more likely to be taught in a mainline computer science textbook on algorithms [19], than a natural language processing (NLP) textbook in computational linguistics (CL) [23], information retrieval (IR), [4] [22], and machine translation (MT) [8]. At the other end of the continuum, we have grammar checking, which is clearly in the realm of natural language processing, perhaps solely because it has yet to be "solved" to the same extent as simple string matching. This article will focus on techniques in the middle like spelling correction. We have less to say about techniques that are better understood because they have been discussed elsewhere many times before, and techniques that are less well understood because they are less well established in the commercial marketplace.

2.1 Spelling Correction

Most systems start by deciding whether or not a string is in the dictionary. Although there are a number of well known solutions to the dictionary access problem such as the one that has been used in Unix for many years [17], dictionary access continues to be an active area of research and commercial opportunity, as will be discussed in the next subsection.

However, the real challenge for a spelling correction system is what to do when the string is not in the table. Most systems assume that the string is a typographic error if it is not in the dictionary, and then they propose a set of "nearby" candidates sorted by some notion of "closeness." It is still an open question what counts as nearby and what counts as closeness. Edit distance (the number of insertions, deletions, substitutions and reversals) is commonly used, though systems are beginning to strive toward some more appropriate, though less clearly defined metrics, such as keyboard distance and phonetic distance. Even the use of the dictionary raises a number of open questions. Just because the string is not in the dictionary does not mean that it is an error; there are lots of legitimate words such as proper nouns and technical terms that will not be found in most state-of-the-art dictionaries. See [18] for a recent survey of research in spelling correction.

2.2 Dictionary Access

Spelling correctors, hyphenation routines and grammar checkers all make heavy use of dictionaries. Different applications require different trade-offs between time and space. A dictionary for a hand-held device should be designed to consume as little memory as possible because the cost of the product is dominated by the number of memory chips. Other applications such as a spelling corrector for use in a WYSIYG (what you see is what you get) editor might require more speed to achieve interactive performance, especially for large documents.

Kaplan and Kay have been working on these issues for over a decade at Xerox PARC [3]. Their work started with a theoretical desire to model a wide variety of morphological and lexical phenomena, ranging from adding an “s” to the end of a word in English to vowel harmony in Turkish, but evolved into a practical set of algorithms based on finite-state transducers, a class of finite-state automata that efficiently represent functions from strings to strings.

Transducer techniques, along with other advances, have been recently used in an internal “help desk” application by Xerox, to facilitate the retrieval of answers to service repair questions from a text database of repair manual and previously answered queries. It also can be found in information retrieval products such as Xerox XSoft’s Visual Recall, and OCR products such as Xerox Imaging Systems’ Textbridge. Finally, Xerox just launched a new line of lexical products through their Desktop Document Systems (DDS) division. DDS is using the technology to create a range of multilingual components that can be embedded in information retrieval, translation, and other document management applications.

This technology, and its first applications (such as the spelling checker for the Xerox MemoryWriter typewriter) were the basis for a startup company called Microlytics, formed in 1985 and later (1987) merged into the publically traded Selectronics Corp. Getting its start with a hardware spell checking device that plugged into typewriters and later keyboards, Microlytics expanded with its own algorithms into such markets as pocket-sized traveler aids based on the Berlitz Dictionary Interpreter, offering a translator that translates 12,000 words to/from any of five languages (or a credit-card sized device that translates 1,000 words to/from any of 26 languages. Through Microlytics, the original Kaplan and Kay algorithms found their way into spelling checkers and thesaurii, such as those included in such popular systems as Micropro, Claris, MacWrite II, Microsoft Word 4 (just the thesaurus), Symantec, etc, and WordFinder software sold to the PC and Apple Macintosh user community. The hand-held language-related device market is now dominated by such companies as Casio, Seiko, Fuji Xerox, Eurotronics, Franklin, Sharp and other primarily Asian manufacturers.

2.3 Localization

Monolingual speakers of English (like the authors) sometimes forget that their language is not the only language in the world. The major word processing applications are being sold throughout Europe and much of the rest of the world. The overseas markets are large and growing. To be successful in these important markets, products have to be *localized* so that they conform to the language and cultural norms of the target customers.

For many software applications, localization can be relatively straightforward. The bulk of the software can be kept more or less intact. Of course, menu options, error messages, help screens and other text strings embedded in the code will have to be translated, but this need not be too hard, especially if the application is designed with localization in mind. As much as possible, the code and the text should be separated. A standard convention is to put text strings in a separate file called a *resource file*. This way, the strings can be translated without touching the source code.

These conventions have been fairly effective for many PC-based applications such as a typical spreadsheet program, but spelling correctors and grammar checkers are much more difficult to localize. Large special-purpose dictionaries may be required. The algorithms may need to be completely redesigned. Software publishers like Microsoft don't always do their own translations and localizations, even for easy applications. They almost certainly don't want to develop spelling correctors and grammar checkers for all of the world's languages. A number of smaller companies such as Infosoft (formerly a division of Houghton-Mifflin), CircleNoetics, Alda Technology and Microlytics are helping to fill many of the gaps.

Localization will continue to offer a number of commercial opportunities for NL technology. The future for localization appears especially promising given the recent emphasis on globalization and multilingualism.

2.4 Internationalization and Translation Aids

In a seamless computing environment for the multilingual world, a software program should be *internationalized*; that is, the user shouldn't have to buy one version of the program for one language and a different version for another. While we may be a ways from achieving this lofty goal, Europe has been working toward an internationalized word processing environment. About 100 million ECU have been invested in the Eurolang Optimizer, produced by the Sietec division of Siemens. It is intended for bilingual secretaries, translators and anyone else who routinely edits text in a variety of languages. It will soon be offered with an interface to Siemens' Metal machine translation program. The version currently on the market is fully integrated into Microsoft's Word-for-Windows, WordPerfect, Frame and a number of other popular word processing applications.

The Eurolang Optimizer, as well as some smaller efforts such as IBM's TranslationManager/2 and Trados' Translation Workbench, offers a number of glossary or definition access and translation-reuse features that may be particularly attractive to translators. Translators know that they need help with technical terminology. How would Microsoft or some other software vendor want "dialog box" to be translated in their manuals? The answers can be surprising. It happens that "dialog box" is translated as *finestra* (window) in Italian and as a *boîte* (box) in French. Translators have trouble with terminology because they are not as familiar with the subject area as either the author of the source text or the readers of the target text. Terminology mistakes are more embarrassing than spelling errors. Spelling errors can be dismissed as mere "typos," but terminology errors make it all too clear that the translator is not an expert in the subject area. Translation schools teach their students to build domain-specific custom glossaries to ensure consistent and correct usage of difficult terminology. Consistency is almost more important than correctness; it would be very confusing if a document translated the word "exit" as *sortir* on one page, and as something else on the next page.

Eurolang offers a number of glossary-access features to help translators make better use of their glossaries. These glossary features are completely integrated into the Eurolang environment. Color is used to help the translator quickly spot terms that are in the glossary. Hot keys and menu items are configured to make it easy for the translator to choose the appropriate equivalent in the target language.

There is also a translation re-use capability which is intended to make it easy for a translator to translate just the changes, and not the entire job. Re-use is important for manuals and other large jobs that are updated on a regular basis and don't change all that much from one version to the next. From a commercial point of view, translation re-use might be even more important than machine translation. At best, machine translation might be able to speed up a translator by a factor of two, whereas translation re-use can achieve much larger speed-ups when there aren't too many changes, which is often the case.

Most translation re-use products use *translation memories* to store sentences that have already been translated so that if the system should encounter the same sentence in the future, it can automatically insert the appropriate translation into the target text. Translation memory tools also provide some sort of fuzzy match facility for sentences that are almost the same as some sentence that has been previously translated.

The potential for a fully-internationalized multilingual word processing environment is very large. Eurolang's message is particularly appealing in Europe, where the new European Union is in the processing of learning how to conduct business in all of their official languages, along with quite a number of unofficial ones as well.

2.5 Controlled Language

Controlled language is another solution for coping with the realities of a multilingual user population. Almost all controlled languages constrain the vocabulary. Many also restrict the grammar. Controlled languages impose the same kinds of standards found in good technical writing: avoid passives, negatives, excessively long sequences of nouns, inconsistent terminology, etc. Controlled languages attempt to improve the clarity of the source document, which is worth-while in its own right. In addition, it is believed that documents written in controlled languages are easier for translators because there is less room for ambiguity. It is also believed that they are easier for non-native speakers for many of the same reasons. Governments, especially in Europe, are beginning to introduce regulations that require the use of controlled languages in international commerce.

Boeing, Caterpillar and a number of other companies have demonstrated a significant commitment to controlled languages. Boeing has deployed the Boeing Simplified English Checker to help ensure that its technical writers conform to the controlled language as defined by various industry standards and government regulations. Caterpillar is working with Carnegie-Mellon University and Carnegie Group to develop an authoring tool called Clearcheck. It is hoped that the Caterpillar method will make it possible to translate manuals with virtually no human post-editing.

In summary, word processing has created a whole new industry worth billions of dollars. The word processing market has become extremely competitive. The major vendors are scrambling to add natural language features like spelling correction, hyphenation and grammar checking. Recently much of the activity has turned to European languages, because the overseas markets are becoming more and more important. The future for localization and internationalization appear promising, given the emphasis on globalization and the multilingualism.

3 Information Management

The introduction cited word processing and information management as two of the better examples of commercial opportunities for natural language processing. The importance of information management is beginning to be appreciated as vast quantities of text become available in electronic form: digital libraries, private databases and even personal email and faxes. The U.S. produces over 2.7 billion sheets of computer printout daily, according to a recent Gartner Group report (November 1994).

Digital libraries are extremely valuable. Lexis-Nexis (formally Mead Data Central), for example, was recently purchased for $1.5 billion. Their main asset is a large collection of text (a half a terabyte $\approx 10^{11}$ words). It wasn't all that long ago that the researchers referred to the Brown Corpus [16] as a "large" corpus. The Brown Corpus, a mere million words collected at Brown University in the 1960s, is about the same size as a dozen novels, the complete works of William Shakespeare, the Bible, a collegiate dictionary or a week of a newswire service. Today, Dialog, Westlaw, Lexis-Nexis and other major vendors of on-line information services are archiving hundreds of megabytes a night, the equivalent of a Brown Corpus an hour.

Many information needs serve government functions, from security, military, financial, and political perspectives. Other civil information needs arise from requirements for processing information at a country-wide level, such as patents, tax returns, and a myriad of forms and filings, even electronic mail to the White House. Public funding of new technologies are supporting a significant number of research and development efforts in both the public and private sector. The political support behind the creation and exploitation of the National Information Infrastructure is providing additional impetus for the design and development of novel applications of information management, many of which will involve some sort of natural language processing.

But you don't need to be a large corporation or government organization to have a problem with information overload. Disks are getting so big, even on a PC, that one can easily misplace an important file. Now that disks are cheaper than real estate (a million words requires about 3 megabytes of disk space after compression, which currently costs about $1, less than the cost of a foot of a book shelf in a library[1] or an office or a private home), the need for better information management tools is clear. Apple, Microsoft, Xerox, and others are working on solutions for "small" computers (which are no longer all that small).

Four types of information management solutions will be discussed:

1. *Retrieval*: Retrieve documents that match a query (or user profile).

2. *Categorization*: Categorize documents into bins.

3. *Extraction*: Extract structured data (e.g., references to people, places, organizations, dates, citations) from natural language.

4. *Generation*: Generate natural language from structured data.

The retrieval and categorization tasks are similar to one another, though the emphasis is on relevant documents in the first case, and on categories in the second case. Extraction and generation can be viewed as inverses of one another. Extraction maps natural language into structured data, and generation reverses the mapping.

3.1 Retrieval

Retrieval systems are typically used to retreive text, though multimedial will undoubtably become more important. It is hard to imagine how a lawyer can read a patent without the figures, but currently, many systems don't offer figures, let alone pictures. PNI (Picture Network International) has just released a product (*Seymour*^{TM}) that matches natural language queries with a database of over 250,000 natural language descriptions of photographs. This product serves

[1]According to Mike Lesk of Bellcore, Cornell's new underground bookstack cost $20 per book, and Berkeley's cost $30 (due to earthquakes). The Harvard Depository costs only $2 per book, much less than on-site underground libraries, but considerably more than disks.

as an automated agency to help freelancers sell photographs to major magazines and other important clients. In the future, we will hear more and more about systems that retrieve World Wide Web (WWW) home pages, videos, and other formats that are popular on the Internet.

As in the previous discussion of Figure 1, the technologies behind retrieval solutions vary from the well understood to the forward looking. Traditionally, most information retrieval products were based on boolean combinations of keywords. The user types in a set of keywords and the system retrieves a set of matching documents. Although these keyword products have helped create a billion dollar on-line services industry,[2] and they continue to dominate the marketplace, there have always been concerns (confirmed by years of controlled experiments [24]) that it is difficult to compose effective keyword queries, especially for novice users.

There have been a number of new products in recent years that attempt to replace or augment keyword systems with statistics and other kinds of natural language processing. Almost all systems (including keyword systems) make use of simple NL techniques such as word stemming, sentence boundary detection, acronym expansion, etc. Some products such as Clarit (Claritech Corp.), Conquest, and the soon-to-be-released ConText (Oracle, formerly Artificial Linguistics Inc.) claim that much of their added value comes from considerably more sophisticated uses of natural language processing. Decades of controlled experiments, though, have found little evidence that NLP improves *precision* (fraction of retrieved documents that are relevant) and *recall* (fraction of relevant documents that are retrieved) over purely statistical methods.

The statistical approach (with varying amounts of NLP) has received considerable attention with the succuss of Westlaw's WIN (Westlaw is Natural), winner of the 1993 Online Product of the Year award. WIN was so successful that Dialog and Lexis-Nexis responded by releasing *Target* and *Freestyle*. These systems take a query, ideally an arbitrary piece of natural language text (though not quite, especially in the case of Target), and sort the documents in the library by a measure of how much they match the query. The *cosine* measure, for example, simply counts the number of words found in both the query and the document, and normalizes by the length of the query and the length of the document. It is common to weight words inversely by the number of documents that they appear in (inverse document frequency (IDF) weighting). Documents are represented as a vector of words and their frequencies, an unordered "bag of words," ignoring linguistic dependencies on word order such as phrase structure and discourse structure.

Examples of statistical and keyword queries are presented in [24]. The two Nexis versions of their question #6, *find information about employment discrimination against gays*, are:

[2] Most of the value of the on-line service industry resides in the data; text retrieval tools are worth less than $300 million, according to Intelligent Document Management Vista, Gartner Group/New Science, November, 1994.

Freestyle: find information about employment discrimination against gays job work workplace lesbians homosexuals

Boolean: employment or job or work or workplace w/4 discriminat! w/50 gay or lesbian or homosexual

Natural language input is attractive to novices because they don't have to learn an artificial query language: boolean operators (*and, or*), proximity (*w/50*), truncation (*discriminat!* matches *discriminate*, *discrimination*, *discriminant*, etc.). Experts, on the other hand, often prefer to stick with boolean, because they know how to use promixity and other operators to capture linguistic dependencies that would be missed by the "bag of words" model, and they believe these dependencies are important (though advocates of purely statistical methods believe otherwise).

3.2 Categorization

Categorization systems [5] input a large stream of documents, e.g., trouble tickets, CASREPS (miliary casualty reports), intelligence intercepts, newswires, marketing data, etc., and assign them to a relatively small number of predefined categories or indices. The Carnegie Group's Construe system [6], for example, inputs Reuters articles and replaces much of the work that used to be done by a staff of human indexers at Reuters, saving $750,000 in 1990, and even more in subsequent years. AT&T and others have been using categorization systems to route trouble tickets to the appropriate desk for corrective action. In some applications, it isn't necessary to assign each and every document to a category, but merely estimate summary statistics such as the rate of trouble tickets per category per month. Summary statistics such as these can be used to determine whether an attempt to fix a problem (by changing a process in a factory, for example), actually had the desired effect or not. Some categorization systems also attempt to identify so-called "emerging issues," hot topics that may not fit neatly into the current set of predefined categories.

There is considerable debate over the proper role between statistics and knowledge-engineering (building specialized rules by hand). The problem with knowledge-engineering is that it can be very expensive. The Carnegie Group found the ten-person years that they invested in Construe could only be justified because Reuters had a very large volume of input documents. They would not recommend the same approach for small volume applications. Statistical approaches, on the other hand, also have their limitations. They often require massive quantities of labeled training data which may not be available, or may be prohibitively expensive. In a few applications, it has been possible to use an unsupervisized training method, circumventing the need for labeled training data.

3.3 Extraction

Extraction systems map natural language into a structured database. As with many of the other applications discussed in this paper, the technology behind extraction systems varies considerably. The ultimate goal is to determine *who did what to whom* with high reliability. Many products, though, stop far short of this lofty goal, and still produce substantial value.

A number of systems have been developed to scan a document and identify references to entities, e.g., people, organizations, citations, dates, etc. Major vendors of on-line information services such as Westlaw and Lexis-Nexis, for example, have developed methods to identify and canonicalize references to company names (I.B.M. = IBM), legal citations, etc. The Carnegie Group's *NameFinder* extracts company names using a predetermined list of company names. Synthesis Technologies' *TextMachine* identifies references to people, companies, facilities, organizations, phone numbers, addresses, and social security numbers. Many custom systems have been developed at SRA for government and commercial use, and their generic extraction product ($NameTag^{TM}$) is slated for commercial release mid-1995. GE Aircraft Engines and United Technologies (UT) use extraction technology to create databases of records from service-related documents. Many other commercial systems utilizing this underlying technology will emerge in the future such as automatically adding hypertext links to the World Wide Web (WWW), or visualizing massive text databases.

3.4 Generation

Generation is the inverse of extraction. Generation converts structured data into natural language. Bellcore's PLANDoc system, for example, generates English text from the output of a computer program that determines upgrades to the wiring service of a phone company. In this way, it is hoped that the effort that went into designing the next upgrade can be re-used in the future, facilitating an improved corporate memory for a time-consuming intellectual exercise.

Perhaps the most widely used NL generation system to date is the Forecast Generator (FOG) system, developed for the Canadian Atmospheric Environment Service [7]. FOG is intended to replace the Meteo translation system [8] (chapter 12), which has been translating about 20 million words of weather forecasts per year for over a decade with almost no pre-conditioning and almost no post-editing. Rather than writing the forecasts in one natural language and then translating them to a second, it is more efficient to store the forecasts in a common database and generate both the English and the French directly from the database.

Both Meteo and FOG attribute much of their success to the restricted nature of the target texts; weather reports are a classic example of a "sublanguage" [9]. The limited vocabulary, grammar and semantics makes it possible to achieve excellent results with relatively simple methods.

However, sublanguages also have their critics. Sublanguages often turn out to be far richer and more complicated than anyone could have possibly anticipated. Roger Schank, for example, likes to point out that it took Cognitive Systems (CSI) ten years to build a system to understand a single sentence (and variations thereof): "Please transfer $X to my account number Y." Quite a number of sublanguages have been investigated over the past decade or two, but few if any have been as successful as the weather.

4 Machine Translation

4.1 Excessive Optimism

Of course, along with the successes mentioned thus far, NLP and AI have also produced their share of failures. Even though we should know better, it is so appealing to fantasize about intelligent computers that understand human communication, that hyperbole is practically unavoidable. It is hard to find a brochure for an AI/NLP product without a reference to "intelligence" and "understanding."

Sometimes these practices work out for the best. Symantec, for example, a highly successful vendor of software tools for the PC, started with a product called Q&A, an NLP program for querying a database. Gary Hendrix, a founder of Symantec, believes Q&A was successful because of its unique packaging of AI/NLP with a good simple database facility. Neither would have been successful in isolation. The AI/NLP generated initial sales, but the real value was in the database. People bought the product because they were intrigued with the AI/NLP technology, but most users ended up turning off the AI/NLP features[3].

But all too often excessive optimism results in a manic-like cycle of euphoric activity followed by severe depression. Perhaps the worst example of a success catastrophe was the so-called 1954 Georgetown University Experiment. In 1954, Georgetown University demonstrated what would now be called a "toy" system. It was designed to translate a small corpus of roughly fifty Russian sentences into English. Little if any attempt was made to generalize to sentences beyond the tiny test corpus.

Unfortunately, these kinds of demos can be extremely convincing, often too convincing. People want to believe in the machine so much that the demo practically sells itself. The 1954 Georgetown experiment received wide publicity.

[3] A similar situation may be unfolding with vendors of new text retrieval products, starting with Topic (Verity) and continuing the trend with Conquest (Conquest) and perhaps next, Oracle's ConText product. The promise of "concept searching" and/or NL querying can sell the product, but the ultimate market success depends on the basic functionality of the software, e.g., operating environment, support for client/server operation, scalability, ability to perform relevance ranking of documents etc.

The "experiment" was summarized by Dostert as "an authentic machine translation which does not require pre-editing of the input nor post-editing of the output" [25] (p. 29)

The following decade was an exciting time for MT. Systran, one of the better systems on the market today, is a by-product of this period. But the euphoria could not last for long. Too much had been promised.

> "The development of the electronic digital computer quickly suggested that machine translation might be possible. The idea captured the imagination of scholars and administrators. The practical goal was simple: to go from machine-readable foreign technical texts to useful English text, accurate, readable and ultimately indistinguishable from text written by an American scientist. Early machine translations of simple or selected text... were as deceptively encouraging as 'machine translations' of general scientific text have been uniformly discouraging." [1] (pp. 23-24)

In addition, the ALPAC report suggested that investment in MT did not make economic sense (p. 29):

> Over the past 10 years the government has spent, through various agencies, some $20 million on machine translation and closely related subjects (see Appendix 16). This is more than the government cost of translation for one year.

Excessive optimism continues to generate wild funding cycles. There were great hopes for MT in Japan in the late 1980s. In 1989, as the Japanese were beginning to invest heavily in machine translation research, they published the JEIDA [10] report, a response to the ALPAC report. JEIDA rejected the economic arguments in the ALPAC report as dated and inapplicable to the situation in Japan. The size of the Japanese translation market was estimated at 800 billion yen ($8 billion). This is a fantastic sum[4]. Even if one adjusts for thirty years of inflation, and the differences between the US and Japan, it still isn't possible to reconcile the two estimates. The truth is probably somewhere in between. Prediction errors can be costly. The ALPAC's low estimate has almost certainly contributed to missed opportunities. Just as certainly, though, the high estimate in the JEIDA report is at least partly responsible for the boom and bust in MT research in Japan in recent years.

4.2 Successes: Large and Small

Of course, we don't want to dwell too much on the failures, either. As we mentioned, there have been a number of successes especially in word processing

[4] Eurolang's advertizing material estimates the world wide translation market at $12 billion. It is unlikely that two thirds of the world-wide translation market is in Japan.

and information management, where whole new industries are being created. We have also mentioned quite a number of successful small projects that are generating a few million or perhaps even a few tens of millions of dollars a year, more than enough to support a small company.

Research and development activities in text processing and interpretation have grown over the past few years, and the ARPA Human Language Technology program (HLT) has been the current focal point for much advanced research and development in the area [12]. This program funds leading-edge work in speech recognition, machine translation, information extraction from text, and text retrieval, in addition to a host of smaller strategic thrusts. Under the HLT umbrella, various benchmarking activities are regularly held. Much of the research activities are just now making their way into the commercial marketplace, as the technology matures. Greater detail on these and related activities can be found in the specific proceedings from these evaluations [13, 14].

Even in the machine translation area, an area that has been having a hard time living down the stigma of the ALPAC report, there are plenty of examples of successful small business opportunities. Systran, for example, is one of the oldest machine translation systems on the market, and continues to be one of the best. With its massive lexicons, it almost always outperforms even the most modern research systems, as reported in the ARPA MT evaluations.

A number of companies such as Globalink and Microtac (which recently merged into a single company) are experiencing rapid growth[5] by opening up new markets for Systran-like MT technology. They have been advertizing PC-based and hand-held MT products in airline magazines, targeting casual users and international travelers, a larger market than professional translators. These products are priced competitively with textbooks for learning a new language, and probably offer better value for the money. Clearly, there have been many highly successful machine translation products.

A completely different kind of argument is often used to justify a major high-profile research and development program, especially in some parts of some large companies and government institutions. Two of these "big success" arguments are presented in the ALPAC report, which was chaired by John Pierce, a former executive of AT&T. These arguments helped to generate an increase in support for computational linguistics, though they also responsible, at least in part, for a decrease in support for machine translation.

1. Appendix 18 noted that context-free grammars (CFGs), then known as Type 2, were helping to create a new industry in software. CFGs had a significant impact on ALGOL-60, an innovative language that introduced block-structure and Backus-Naur form (BNF), a context-free-like notation for specifying the syntax of a programming language. These innovations can be found in practically every modern programming language, demonstrating the value of long-term fundamental research.

[5] Globalink sold $20 million in 1994, probably more than any other vendor of MT software.

2. Appendix 17 mentioned computerized publishing, which has since mushroomed into the word processing and desktop publishing industry. In fact, over the past forty years, this application may have become even more important than programming. We used to be embarrassed to admit that we were using our million dollar computers to emulate a hundred dollar typewriter. Computers were supposed to be used for writing programs, not for writing prose. But the situation has changed now that computers have become a commodity. Word processing applications are out-selling compilers (developer's kits) because everyone writes prose, but only a few write programs. Now that everyone is doing it, we no longer need to be embarrassed to admit that computers make really good typewriters.

If the ALPAC committee were to rewrite their report today, they might add a few more items to their list of "big success" arguments. Something would have to be said about the so-called "information highway," given all the fuss surrounding that topic. Information management and text retrieval have become much more important in recent years, and we expect the trend to continue with multimedia systems.

5 Conclusions

In this article, we have sought to demonstrate the real and potential profitability of natural language processing systems today, while keeping the lessons learned from the past in mind, and the dangers (and occasionally, opportunities) of "hype" at bay. We have mentioned quite a number of successes, both large and small. Systran is one of the oldest MT systems on the market and still one of the best. Meteo is translating 20 million words a year with almost no human intervention, saving the Canadian government a few million dollars per year. Spelling correctors, hyphenation routines and grammar checkers are becoming more and more standard in the lucrative word processing market. There will continue to be many opportunities to improve and extend these word processing features, especially as a result of localization and internationalization efforts. There are also many opportunities for natural language technology in information management, which is becoming increasingly important with the availability of vast quantities of text in electronic form.

6 Acknowledgements

Many people contributed to the source material for this article, only a small fraction of which could be included. Three anonymous reviewers provided valuable feedback as well. Further information on the applications and services mentioned throughout can be obtained from the contact points as follows:

Alda Technology: Jargon Product (905-829-3461); **Apple Research**: Branimir Boguraev (bjr@apple.com); **Association for Machine Translation in the Americans**: (655 15th St NW, Suite 310, Washington, DC 20005); **Boeing Advanced Technology Center**: James Hoard (jhoard@grace.boeing.com); **Bellcore**: Karen Kukich (kukich@bellcore.com); **Carnegie Group**: Phil Hayes (412-642-6900, hayes@cgi.com); **CircleNoetics**: Gillian Smith (603-672-6151); **Claritech**: David Evans (dae@clarit.com, 412-268-8574); **CoGenTex**: Richard Kittredge (kittredg@IRO.UMontreal.CA); **Cognitive Systems, Inc**: Steven Mott (203-356-7756); **ConQuest**: Ed Addison (); **Eurolang Optimizer**: Lutz Graunitz (800-565-5650, lutz@sni.ca); **Globalink**: Brian D. Stagger (800-255-5660); **IBM**: Roy Byrd (byrd@watson.ibm.com); **InfoSoft**: Win Carus (carus@hmco.com); **Lexis-Nexis**: Fu-qiu Zhou (joez@meaddata.com); **Microlytics/Selectronics**: Mike McCourt (716-248-9150); **Microsoft**: Karen Jensen (karenje@microsoft.com); **PNI**: Sharon Flank (703-558-8455); **SRI**: Jerry Hobbs (hobbs@ai.sri.com); **Synthesis Technologies**: R. Daniel Robinson (800-695-9857); **Systran Translation Systems, Inc.**: Ana Carder (619-459-6700) **TextWise**: Michael Weiner (315-443-2911); **United Technologies**: Benjamin Moreland (bjm@utrc.utc.com, 203-727-7729); **West Publishing**: Howard Turtle (turtle@research.westlaw.com, 617-687-5660); **WordPerfect**: Blake Stowell (801-225-5000); **Xerox PARC**: Ronald Kaplan (Ronald_Kaplan.PARC@xerox.com):

References

[1] ALPAC. Languages and machines: Computers in translation and linguistics. Report of the Automatic Language Processing Advisory Committee 1416, Division of Behavioral Sciences, National Academy of Sciences, Washington, D. C., 1966.

[2] Karen Jensen, George Heidorn, and Stephen Richardson. *Natural Language Processing: The PLNLP Approach*. Kluwer Academic Publisher, Boston, 1993.

[3] Ronald M. Kaplan and Martin Kay. Regular models of phonological rule systems. *American Journal of Computational Linguistics*, 1994.

[4] G. Salton and M. Mcgill. *An Introduction To Modern Information Retrieval*. Mcgraw-Hill, New York, 1983.

[5] David D. Lewis and Philip J. Hayes (editors). Special issue on text categorization. *ACM Transations on Information Systems*, 12(3):231–334, July 1994.

[6] Philip J. Hayes and Steven P. Weinstein. CONSTRUE/TIS: A system for content-based indexing of a database of news stories. In *Proceedings*

of the Second Annual Conference on Innovative Applications of Artificial Intelligence, pages 49–64, May 1991.

[7] Eli Goldberg, Norbert Driedger, and Richard Kittredge. Using natural-language processing to produce weather forecasts. *IEEE Expert*, 9(2):45–53, April 1994.

[8] W. John Hutchins and Harold L. Somers. *An Introduction to Machine Translation*. Academic Press, London, 1992.

[9] Richard Kittredge. Variation and homogeneity of sublanguages. In Richard Kittredge and John Lehrberger, editors, *Sublanguages: Studies of Language in Restricted Domains*, pages 107–137. Walter DeGruyter, New York, 1982.

[10] Makoto Nagao (Chairman), The Machine Translation System Research Commitee A japanese view of machine translation in light of the considerations and recommendations reported by ALPAC, USA. Technical report, Japan Electronic Industry Development Association, Japan, 1989 July.

[11] John S. White, Theresa A. O'Connell, and Lynn M. Carlson. Evaluation of machine translation. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 206–210, Princeton, NJ, March 1993. Morgan Kaufmann.

[12] *Proceedings of the ARPA Speech and Natural Language Workshop*. Morgan Kaufmann, Princeton, NJ, March 1994.

[13] Beth Sundheim, editor. *Proceedings of the Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann Publishers, San Mateo, Ca, June 1994.

[14] Donna Harman, editor. *Proceedings of the Third Text Retrieval Conference (TREC)*. Morgan Kaufmann Publishers, San Mateo, Ca, February 1995.

[15] Paul S. Jacobs. *Proceedings of the Fourth Applied Conference on Natural Language Processing*. 1995.

[16] W. Francis and H. Kucera. Houghton Mifflin, Brown University, 1982.

[17] M. McIlroy. Development of a Spelling List *IEEE Trans. on Communications*, 30: 91–99, 1982.

[18] K. Kukich. Techniques for Automatically Correcting Words in Text *ACM Computing Surveys*, 24:4, 1992.

[19] T. Cormen, C. Leiserson, R. Rivest. *Introduction to Algorithms* MIT Press, Cambridge, MA, USA, 1990.

[20] W. Teitelman, *et al.* InterLISP Reference Manual, Third Revision, Xerox Palo Alto Research Center, Palo Alto, CA, USA, 1978.

[21] W. Woods, M. Bates, *et al.* Speech understanding research at BBN, Final report on natural communication with computers, BBN report 2976, 1974.

[22] W. Frakes and R. Baeza-Yates (eds). *Information Retrieval: Data Structures & Algorithms* Prentice Hall, New Jersey, USA, 1992.

[23] J. Allen. *Natural Language Understanding* The Benjamin/Cummings Publishing Company, Inc. Menlo Park, CA, USA, 1987.

[24] Carol Tenopir and Pamela Cahn Target & Freestyle: Dialog and Mead Join the Revance Ranks *Online* pp. 31-44, May 31, 1994.

[25] M. Zarechnak "This History of Machine Translation," in Bozena Henisz-Dostert, R. Ross Macdonald, Michael Zarechnak (eds), *Machine Translation*, Mouton Publishers, 1979.