Computational Linguistics and Chinese Language Processing
vol. 3, no. 2, August 1998, pp. 1-16
©Computational Linguistics Society of R.O.C.

Senses and Texts

Yorick Wilks*

Abstract

This paper addresses the question of whether it is possible to sense-tag systematically, and on a large scale, and how we should assess progress so far. That is to say, how to attach each occurrence of a word in a text to one and only one sense in a dictionary---a particular dictionary of course, and that is part of the problem. The paper does not propose a solution to the question, though we have reported empirical findings elsewhere [Cowie et al. 1992 and Wilks et al. 1996], and intend to continue and refine that work. The point of this paper is to examine two well-known contributions critically, one [Kilgarriff 1993] which is widely taken as showing that the task, as defined, cannot be carried out systematically by humans, and secondly [Yarowsky 1995] which claims strikingly good results at doing exactly that.

1. Introduction

Empirical, corpus-based, computational linguistics reached by now into almost every crevice of the subject, and perhaps pragmatics will soon succumb. Semantics, if we may assume the sense-tagging task is semantic, taken broadly, has shown striking progress in the last five years and, in Yarowsky's most recent work [1995] has produced very high levels of success in the 90%s, well above the key bench-mark figure of 62% correct sense assignment, achieved at an informal experiment in New Mexico about 1990, in which each word was assigned its FIRST sense listed in LDOCE.

A crucial question in this paper will be whether recent work in sense-tagging has in fact given us the breakthrough in scale that is now obvious with, say, part-of-speech tagging. Our conclusion will be that it has not, and that the experiments so far, however high their success rates, are not yet of a scale different from those of the previous generation of linguistic, symbolic-AI or connectionist approaches to the very same

*Department of Computer Science, University of Sheffield, 211 Portobello Street, Sheffield, S1 4DP, UK.
E-mail: yorick@dcs.sheffield.ac.uk

problem.

A historian of our field might glance back at this point to, say, Small et al. [1988] which covered the AI-symbolic and connectionist traditions of sense-tagging at just the moment before corpus-driven empirical methods began to revive. All the key issues still unsettled are discussed there and the collection showed no naivet there about the problem of sense resolution with respect only to existing lexicons of senses. It was realised that that task was only meaningful against an assumption of some method for capturing new (new to the chosen lexicon, that is) senses and, most importantly, that although existing lexicons differed, they did not differ arbitrarily much. The book also demonstrated that there was also strong psychological backing for the reality of word senses and for empirical methods of locating them from corpora without any prior assumptions about their number or distribution [e.g. Plate's work in Wilks et al. 1990, and see also Jorgensen, 1990].

Our purpose in this paper will be to argue that Kilgarriff's negative claims are simply wrong, and his errors must be combated, while Yarowsky is largely right although we have some queries about the details and the interpretation of his claims. Both authors however agree that this is a traditional and important task: one often cited as being, because of the inability of systems of the past to carry it out, a foundational lacuna in, say, the history of machine translation (MT). It was assumed by many, in that distant period, that if only word-sense ambiguity could be tamed, by the process we are calling sense-tagging, then MT of high quality would be relatively straightforward. Like may linguistic tasks, it became an end in itself, like syntactic parsing, and , now that it is, we would claim, firmly in sight (despite Kilgarriff) it is far less clear that its solution will automatically solve a range of traditional problems like MT. But clearly it would be a generally good tool to have and local triumph if this long-resistant bastion of NLP were to yield.

2. The very possibility of sense-tagging

Kilgarriff's paper [1993] is important because it has been widely cited as showing that the senses of a word, as distinguished in a dictionary such as LDOCE, do not cover the senses actually carried by most occurrences of the word as they appear in a corpus. If his paper does show that, it is very significant indeed, because that would imply that sense-tagging word occurrences in a corpus by means of any lexical data based on, or related to, a machine-readable dictionary or thesaurus is misguided. I want to show that here the paper does not demonstrate any such thing. Moreover, it proceeds by means of a straw-man it may be worth bringing back to life!

That straw-man, Kilgarriff's starting point, is the 'bank model' (BM) of lexical ambiguity resolution, which is established by assertion rather than quotation, though it is attributed to Small, Hirst, and Cottrell as well as the present author. In the BM, words have discrete meanings, and the human reader (like the ideal computer program) knows instantly and effortlessly which meaning of the word applies [Ibid. p.367], "given that a word occurrence always refers to one or the other, but not both" of the pair of main meanings that a word like 'bank' is reputed to have. The main thrust of Kilgarriff's paper is to distinguish a number of relationships between LDOCE senses that are not discrete in that way, and then to go on to an experiment with a corpus.

But first we should breathe a little life back into the BM straw-man: those named above can look after themselves, but here is a passage from Wilks [1972, p.12] "..it is very difficult to assign word occurrences to sense classes in any manner that is both general and determinate. In the sentences "I have a stake in this country" and "My stake on the last race was a pound" is "stake" being used in the same sense or not? If "stake" can be interpreted to mean something as vague as "Stake as any kind of investment in any enterprise" then the answer is yes. So, if a semantic dictionary contained only two senses for "stake": that vague sense together with "Stake as a post", then one would expect to assign the vague sense for both the sentences above. But if, on the other hand, the dictionary distinguished "Stake as an investment" and "Stake as an initial payment in a game or race" then the answer would be expected to be different. So, then, word sense disambiguation is relative to the dictionary of sense choices available and can have no absolute quality about it". QED.

In general, it is probably wise to believe, even if it is not always true, that authors in the past were no more naive than those now working, and were probably writing programs, however primitive and ineffective, to carry out the very same tasks as now (e.g. sense-tagging of corpus words). More importantly, the work quoted, which became an approach called preference semantics, was essentially a study of the divergence of corpus usage from lexical norms (or preferences) and developed in the Seventies into a set of processes for accommodating divergent/non-standard/metaphorical or what-you-will usage to existing lexical norms, notions that Kilgarriff seems to believe only developed in a much later and smarter group of people around 1990, which includes himself, but also, for example, Fass whose work was a direct continuation of that quoted above. Indeed, in Wilks [1972] procedures were programmed (and run over a set of newspaper editorials) to accommodate the divergent usage to that of an established sense of another word in the same text, while in Wilks [1978] programmed procedures were specified to accommodate such usage by constructing completely new sense entries.

A much more significant omission, one that bears directly on his main claim and is not merely an issue of historical correctness, is the lack of reference to work in New Mexico and elsewhere [e.g. Cowie et al. 1992] on the large-scale sense tagging of corpora against an MRD-derived lexical data base. These were larger scale experiments whose results directly contradict the result he is believed to have proved. I shall return to this point in a moment. The best part of Kilgarriff's paper is his attempt to give an intuitive account of developmental relations between the senses of a word: there is, of course, a large scholarly literature on this. He distinguishes Generalizing Metaphors (a move from a specific case to a more general one), from Must-be-theres (the applicability of one sense requires the applicability of another, as when an act of matricide requires there to be a mother); from Domain shift (where a sense in one domain, like "mellow" of wine, is far enough from the domain of "mellow" of a personality, to constitute a sense shift).

It is not always easy to distinguish the first two types, since both rest on an implication relationship between two or more senses. Again, the details do not matter: what he has shown convincingly is that, as in the earlier quotation, the choice between senses of a given word is often not easy to make because it depends on their relationship, the nature of the definitions and how specific they are. I suspect no one has ever held a simple-minded version of the BM, except possibly Fodor and Katz, who, whatever their virtues, had no interest at all in lexicography.

The real problem with Kilgarriff's analysis of sense types is that he conflates:

a) text usage different from that shown in a whole list of stored senses for a given word e.g. in a dictionary, (which is what his later experiment will be about) with

b) text usage divergent from some "core" sense in the lexicon.

Only the second is properly in the area of metaphor/metonymy or "grinding" [Copestake and Briscoe, 1991] work of the group in which he places himself, and it is this phenomenon to which his classification of sense distinctions summarized above properly belongs. This notion requires some idea of sense development; of senses of a word extending in time in a non-random manner, and is a linguistic tradition of analysis going back to Givon [1967]. However, the straw-man BM and the experiment he then does on hand-tagging of senses in text, all attach to the first, unrelated, notion which does not normally imply the presence of metonymy or metaphor at all, but simply an inadequate sense list. Of course, the two types may be historically related, in that some of the (a) list may have been derived by metaphorical/metonymic processes from a (b) word, but this is not be so in general. This confusion of targets is a weakness in the paper, since it makes it difficult to be sure what he wants us to conclude from the experiment. However, since we shall show his results are not valid, this distinction may not matter too much.

One might add here that Kilgarriff's pessimism has gone hand in hand with some very interesting surveys he has conducted over the Internet on the real need for word-sense disambiguation by NLP R&D. And one should note that there are others [e.g. Ide and Veronis, 1994] who have questioned the practical usefulness of data derived at many sites from MRDs. Our case here, of course, is that it has been useful, both in our own work on sense-tagging [Cowie et al.op.cit.] and in that of Yarowsky, using Roget and discussed below.

Kilgarriff's experiment, which what has been widely taken to be the main message of his paper, is not described in much detail. In a footnote, he refuses to give the reader the statistics on which his result was based even though the text quite clearly contains a claim [p. 378] that 87% of (non-monsemous) words in his text sample have at least one text occurrence that cannot be associated with one and only one LDOCE sense. Hence, he claims, poor old BM is refuted, yet again.

But that claim (about word types) is wholly consistent with, for example, 99% of text usage (of word tokens) being associated with one and only one dictionary sense! Thus the actual claim in the paper is not at all what it has been taken to show, and is highly misleading.

But much empirical evidence tells also against the claim Kilgarriff is believed to have made. Informal analyses [1989] by Georgia Green suggested that some 20% of text usage (i.e. to word tokens) could not be associated with a unique dictionary sense. Consistent with that, too, is the use of simulated annealing techniques by Cowie et al. [1992] at CRL-New Mexico to assign LDOCE senses to a corpus. In that work, it was shown that about 75%-80% of word usage could be correctly associated with LDOCE senses, as compared with hand-tagged control text. It was, and still is, hoped that that figure can be raised by additional filtering techniques.

The two considerations above show, from quite different sources and techniques, the dubious nature of Kilgarriff's claim. Wierzbicka [1989 following Antal 1963] has long argued that words have only core senses and that dictionaries/lexicons should express that single sense and leave all further sense refinement to some other process, such as real world knowledge manipulations, AI if you wish, but not a process that uses the lexicon. Since the CRL result suggested that the automatic procedures worked very well (nearer 80%) at the homograph, rather than the sub-sense, level (the latter being where Kilgarriff's examples all lie) one possible way forward for NLP would be to go some of the way with Wierzbicka's views and restrict lexical sense distinctions to the homograph level. Then sense tagging could perhaps be done at the success level of part-of speech tagging. Such a move could be seen as changing the data to suit what you can accomplish, or as reinstating AI and pragmatics within NLP for the kind of endless, context-driven, inferences we need in real situations.

This suggestion is rather different from Kilgarriff's conclusion: which is also an empirical one. He proposes that the real basis of sense distinction be established by usage clustering techniques applied to corpora. This is an excellent idea and recent work at IBM [Brown et al. 1991] has produced striking non-seeded clusters of corpus usages, many of them displaying a similarity close to an intuitive notion of sense.

But there are serious problems in moving any kind of lexicography, traditional or computational, onto any such basis. Hanks [1994] has claimed that a dictionary could be written that consisted entirely of usages, and has investigated how those might be clustered for purely lexicographic purposes, yet it remains unclear what kind of volume could result from such a project or who would buy it and how they could use it. One way to think of such a product would be the reduction of monolingual dictionaries to thesauri, so that to look up a word becomes to look up which row or rows of context bound semi-synonyms it appears in. Thesauri have a real function both for native and non-native speakers of a language, but they rely on the reader knowing what some or all of the words in a row or class mean because they give no explanations. To reduce word sense separation to synonym classes, without explanations attached would limit a dictionary's use in a striking way.

If we then think not of dictionaries for human use but NLP lexicons, the situation might seem more welcoming for Kilgarriff's suggestion, since he could be seen as suggesting, say, a new version of WordNet [Miller, 1985] with its synsets established not a priori but by statistical corpus clustering. This is indeed a notion that has been kicked around in NLP for a while and is probably worth a try. There are still difficulties: first, that any such clustering process produces not only the clean, neat, classes like IBM's (Hindu Jew Christian Bhuddist) example but inevitable monsters, produced by some quirk of a particular corpus. Those could, of course, be hand weeded but that is not an automatic process.

Secondly, as is also well known, what classes you get, or rather, the generality of the classes you get, depends on parameter settings in the clustering algorithm: those obtained at different settings may or may not correspond nicely to, say, different levels of a standard lexical hierarchy. They probably will not, since hierarchies are discrete in terms of levels and the parameters used are continuous but, even when they do, there will be none of the hierarchical terms attached, of the sort available in WordNet (e.g. ANIMAL or DOMESTIC ANIMAL). And this is only a special case of the general problem of clustering algorithms, well known in information retrieval, that the clusters so found do not come with names or features attached.

Thirdly, and this may be the most significant point for Kilgarriff's proposal, there will always be some match of such empirical clusters to any new text occurrence of a word and, to that degree, sense-tagging in text is bound to succeed by such a methodology, given the origin of the clusters and the fact that a closest match to one of a set of clusters can always be found. The problem is how you interpret that result because, in this methodology, no hand-tagged text will be available as a control since it is not clear what task the human controls could be asked to carry out. Subjects may find traditional sense-tagging (against e.g. LDOCE senses) hard but it is a comprehensible task, because of the role dictionaries and their associated senses have in our cultural world. But the new task (attach one and only one of the classes in which the word appears to its use at this point) is rather less well defined. But again, a range of original and ingenious suggestions may make this task much more tractable, and senses so tagged (against WordNet style classes, though empirically derived) could certainly assist real tasks like MT even if they did not turn out wholly original dictionaries for the book buying public.

There is, of course, no contradiction between, on the one hand, my suggestion for a compaction of lexicons towards core or homograph senses, done to optimize the sense-tagging process and, on the other, his suggestion for an empirical basis for the establishment of synsets, or clusters that constitute senses. Given that there are problems with wholly empirically-based sense clusters of the sort mentioned above, the natural move would be to suggest some form of hybrid derivation from corpus statistics, taken together with some machine-readable source of synsets: WordNet itself, standard thesauri, and even bilingual dictionaries which are also convenient reductions of a language to word sets grouped by sense (normally by reference to a word in another language, of course). As many have now realised, both the pure corpus methods and the large-scale hand-crafted sources have their virtues, and their own particular systematic errors, and the hope has to be that clever procedures can cause those to cancel, rather than reinforce, each other. But all that is future work, and beyond the scope of a critical note.

In conclusion, it may be worth noting that the BM, in some form, is probably inescapable, at least in the form of what Pustejovsky [1995] calls a "sense enumerative lexicon", and against which he inveighs for some twenty pages before going on to use one for his illustrations, as we all do, including all lexicographers. This is not hypocrisy but a confusion close to that between (a) and (b) above: we, as language users and computational modellers, must be able, now or later, to capture a usage that differs from some established sense (problem b above), but that is only loosely connected to problem (a), where senses, if they are real, seem to come in lists and it is with them we must sense-tag if the task is to be possible at all.

3. Recent experiments in sense-tagging

We now turn to the claims in [Gale, Church & Yarowsky 1992, abbreviated to GCY, see also Yarowsky 1991, 1993 and 1995] that:

(1) That word tokens in text tend to occur with a smaller number of senses than often supposed and, most specifically,

(2) In a single discourse a word will appear in one and only one sense, even if several are listed for it in a lexicon, at a level of about 94% likelihood for non-monosemous words (a figure that naturally becomes higher if the monosemous text words are added in).

These are most important claims if true for they would, at a stroke, remove a major excuse for the bad progress of MT; make redundant a whole sub-industry of NLP, namely sense resolution, and greatly simplify the currently fashionable NLP task of sense-tagging texts by any method whatever [e.g. Cowie et al. op cit., Bruce & Wiebe 1994].

GCY's claim would not make sense-tagging of text irrelevant, of course, for it would only allow one to assume that resolving any single token of a word (by any method at all) in a text would then serve for all occurrences in the text, at a high level of probability. Or, one could amalgamate all contexts for a word and resolve those taken together to some pre-established lexical sense. Naturally, these procedures would be absurd if one were not already convinced of the truth of the claim.

GCY's claims are not directly related to those of Kilgarriff, who aimed to show only that it was difficult to assign text tokens to any lexical sense at all. Indeed, Kilgarriff and GCY use quite different procedures: Kilgarriff's is one of assigning a word token in context to one of a set of lexical sense descriptions, while GCY's is one of assessing whether or not two tokens in context are the same sense or not. The procedures are incommensurable and no outcome on one would be predictive for the other: GCYs procedures do not use standard lexicons and are in terms of closeness-of-fit, which means that, unlike Kilgarriff's, they can never fail to match a text token to a sense, defined in the way they do (see below).

However, GCYs claims are incompatible with Kilgarriff's in spirit in that Kilgarriff assumes there is a lot of polysemy about and that resolving it is tricky, where GCY assume the opposite.

Both Kilgarriff and GCY have given rise to potent myths about word-sense tagging in text that I believe are wrong, or at best unproven. Kilgarriff's paper, as we saw earlier, has some subtle analysis but one crucial statistical flaw. GCY's is quite different: it is a mush of hard to interpret claims and procedures, but ones that may still, nonetheless, be basically true.

GCY's methodology is essentially impressionistic: the texts they chose are, of course, those available, which turn out to be Grolier's Encyclopaedia. There is no dispute about one-sense-per-discourse (their name for claim (2) above) for certain classes of texts: the more technical a text the more anyone, whatever their other prejudices about language, would expect the claim to be true. Announcing that the claim had been shown true for mathematical or chemical texts would surprise no one; encyclopaedias are also technical texts.

Their key fact in support of claim (1) above, based on a sense-tagging of 97 selected word types in the whole Encyclopaedia, and sense tagged by the statistical method described below, was that 7569 of the tokens associated with those types are monosemous in the corpus, while 6725 are of words with more than two senses. Curiously, they claim this shows "most words (both by token and by type) have only one sense". I have no idea whether to be surprised by this figure or not but it certainly does nothing to show that [op.cit., 1992] "Perhaps word sense disambiguation is not as difficult as we might have thought". It shows me that, even in fairly technical prose like that of an encyclopaedia, nearly half the words occur in more than one sense.

And that fact, of course, has no relation at all to mono- or poly-semousness in whatever base lexicon we happen to be using in an NLP system. Given a large lexicon, based on say the OED, one could safely assume that virtually all words are polysemous. As will be often the case, GCY's claim at this point is true of exactly the domain they are dealing with, and their (non-stated) assumption that any lexicon is created for the domain text they are dealing with and with no relation to any other lexicon for any other text. One claim per discourse, one might say.

This last point is fundamental because we know that distinctions of sense are lexicon- or procedure-dependent. Kilgarriff faced this explicitly, and took LDOCE as an admittedly arbitrary starting point. GCY never discuss the issue, which makes all their claims about numbers of senses totally, but inexplicitly, dependent on the procedures they have adopted in their experiments to give a canonical sense-tagging against which to test their claims.

This is a real problem for them. They admit right away that few or no extensive hand-tagged sense-resolved corpora exist for control purposes, So, they must adopt a sense-discrimination procedure to provide their data that is unsupervised. This is where the ingenuity of the paper comes in, but also its fragility. They have two methods for providing sense-tagged data against which to test their one-sense-per-discourse claim (2).

The first rests on a criterion of sense distinction provided by correspondence to differing non-English words in a parallel corpus, in their case the French-English Canadian Hansard because, as always, it is there!. So, the correspondence of "duty" to an aligned sentence containing either "devoir" or "impot" (i.e. obligation or tax) is taken as an effective method of distinguishing the obligation/tax senses of the English word, which was indeed the criterion for sense argued for in [Dagan and Itai, 1994]. It has well known drawbacks: most obviously that whatever we mean by sense distinction in English, it is unlikely to be criterially revealed by what the French happen to do in their language.

More relevantly to the particular case, GCY found it very hard to find plausible pairs for test, which must not of course SHARE ambiguities across the French/English boundaries (as interest/interet do). In the end they were reduced to a test based on the six (!) pairs they found in the Hansard corpus that met their criteria for sense separation and occurrence more than 150 times in two or more senses. In GCYs defence one could argue that, since they do not expect much polysemy in texts, examples of this sort would, of course, be hard to find.

Taking this bilingual method of sense-tagging for the six word set as criterial they then run their basic word sense discrimination method over the English Hansard data. This consists, very roughly, of a training method over 100 word surrounding contexts for 60 instances of each member of a pair of senses (hand selected) i.e. for each pair 2x60x100=12,000 words. Notice that this eyeballing method is not inconsistent with anything in Kilgarriff's argument: GCY selected 120 contexts in Hansard for each word that DID correspond intuitively to one of the (French) selected senses. It says nothing about any tokens that may have been hard to classify in this way. The figures claimed for the discrimination method against the criterial data vary between 82 and 100% (for different word pairs) of the data for that sense correctly discriminated.

They then move on to a monolingual method that provides sense-tagged data in an unsupervised way. It rests on previous work by Yarowsky [1991] and uses the assignment of a single Roget category (from the 1042) as a sense-discrimination. Yarowsky sense-tagged some of the Grolier corpus in the following way: 100-word contexts for words like "crane" (ambiguous between bird and machinery) are taken and those words are scored by (very roughly, and given interpolation for local context) which of the 1042 Roget categories they appear under as tokens. The sense of a given token of "crane" is determined by which Roget category wins out: e.g. 348 (TOOLS/MACHINERY) for the machinery contexts, one hopes, and category 414 (ANIMALS/INSECTS) for the bird contexts. Yarowsky [1991] claimed 93% correctness for this procedure over a sample of 12 selected words, presumably checked against earlier hand-tagged data.

The interpolation for local effects is in fact very sophisticated and involves training with the 100 word contexts in Grolier of all the words that appear under a given candidate Roget head, a method that they acknowledge introduces some noise, since it adds into the training material Grolier contexts that involve senses of a category 348 word, say, that is not its machinery sense (e.g. crane as a bird). However, this method, they note, does not have the sense-defined-by-language2 problems that come with the Hansard training method.

In a broad sense, this is an old method, probably the oldest in lexical computation, and was used by Masterman [reported in Wilks 1972] in what was probably the first clear algorithm ever implemented for usage discrimination against Roget categories as sense-criterial. In the very limited computations of those days the hypothesis was deemed conclusive falsified; i.e. the hypothesis that any method overlapping the Roget categories for a word with the Roget categories of neighbouring words would determine an appropriate Roget category for that word in context.

This remains, I suspect, an open question: it may well be that Yarowsky's local interpolation statistics have made the general method viable, and that the 100-word window of context used is far more effective than a sentence. It may be the 12 words that confirm the disambiguation hypothesis at 93% would not be confirmed by 12 more words chosen at random (the early Cambridge work did at least try to Roget-resolve all the words in a sentence). But we can pass over that for now, and head on, to discuss GCY's main claim (2) given the two types of data gathered.

Two very strange things happen at this point as the GCY paper approaches its conclusion: namely, the proof of claim (2) or one-sense-per-discourse. First, the two types of sense-tagged data just gathered, especially the Roget-tagged data, should now be sufficient to test the claim, if a 93% level is deemed adequate for a preliminary test. Strangely, the data derived in the first part of the paper is never used or cited and the reader is not told whether Yarowsky's Roget data confirms or disconfirms (2).

Secondly, the testing of (2) is done purely by human judgement: a "blind" team of the three authors and two colleagues who are confronted by the OALD main senses for one of nine test words, and who then make judgements of pairs of contexts for one of the nine words drawn from a single Grolier article. The subjects are shown to have pretty consistent judgements and, of fifty-four pairs of contexts from the same article, fifty-one shared the same sense and three did not.

Notice here that the display of the OALD senses is pointless, since the subjects are not asked to decide which if any OALD sense the words appear in, and so no Kilgarriff style problems can arise. The test is simply to assign SAME or NOTSAME, and there are some control pairs added to force discrimination in some cases.

What can one say of this ingenious mini-experiment? Lexicographers traditionally distinguish "lumpers" and "splitters" among colleagues: those who tend to break up senses further and those who go for large, homonymic, senses, of which Wierzbicka would be the extreme case. Five GCY colleagues (one had to be dropped to get consistency among the team) from a "lumper" team decided that fifty-one out of fifty-four contexts for a word in a single encyclopaedia article (repeated for eight other words) are in the same sense. Is this significant? I suspect not very, and nothing at all follows to support the myth of discovery that has grown round the paper: the team and data are tiny and not disinterested. The Grolier articles are mini-texts where the hypothesis would, if true, surprise one least. Much more testing is needed before a universal hypothesis about text polysemy enters our beliefs. Of course, they may in the end be right, and all the dogma of the field so far be wrong.

More recently, Yarowsky (1993, 1995) has extended this methodology in two ways: first, he has established a separate claim he calls "one sense per collocation", which is quite independent of local discourse context (which was the separate "one-sense-per-discourse" claim) and could be expressed crudely by saying that it is highly unlikely that the following two sentences (with the "same" collocations for "plants") can both be attested in a corpus:

Plastic plants can fool you if really well made (=organic)

Plastic plants can contaminate whole regions (=factory)

One's first reaction may be to counter-cite examples like "Un golpe bajo" which can mean either a low blow in boxing, or a score one below par, in golf, although "golpe" could plausibly be said to have the same collocates in both cases. One can dismiss such examples (due to Jim Cowie in this case) by claiming both readings are idioms, but that should only focus our mind more on what Yarowsky does mean by collocation.

That work, although statistically impressive, gives no procedure for large-scale sense-tagging taken alone, since one has no immediate access to what cue words would, in general, constitute a collocation sufficient for disambiguation independent of discourse context. An interesting aspect of Yarowsky's paper is that he sought to show that on many definitions of sense and on many definitions of collocation (e.g. noun to the right, next verb to the left etc.) the hypothesis was still true at an interesting level, although better for some definitions of collocation than for others.

In his most recent work [1995] Yarowsky has combined this approach with an assumption that the earlier claim (2: one-sense-per-discourse) is true, so as to set up an iterative bootstrapping algorithm that both extends disambiguating collocational keys [Yarowsky 1993] and retrains against a corpus, while at the same time filtering the result iteratively by assuming (2): i.e. that tokens from the same discourse will have the same sense. The result, on selected pairs (as always) of bi-semous words is between 93 and 97% (for different word pairs again) correct against handcoded samples, which is somewhat better than he obtained with his Roget method (93% in 1991) and better than figures from Schuetze and Pederson [1995] who produce unsupervised clusterings from a corpus that have to be related by hand to intelligible, established, senses. However, although this work has shown increasing sophistication, and has the great advantage, as he puts it, of not requiring costly hand-tagged training sets but instead "thrives on raw, unannotated, monolingual corpora--the more the merrier", it has the defect at present that it requires an extensive iterative computation for each identified bisemous word, so as to cluster its text tokens into two exclusive classes that cover almost all the identified tokens. In that sense it is still some way from a general sense-tagging procedure for full text corpora, especially one that tags with respect to some generally acceptable taxonomy of senses for a word. Paradoxically, Yarowsky was much closer to that last criterion with his 1991 work using Roget that did produce a sense-tagging for selected word pairs that had some "objectivity" predating the experiment.

Although Yarowsky compares his work favorably with that of Schuetze and Pederson in terms of percentages (96.7 to 92.2) of tokens correctly tagged, it is not clear that their lack of grounding for the classes in an established lexicon is that different from Yarowsky, since his sense distinctions in his experiments (e.g. plant as organic or factory) are intuitively fine but pretty ad hoc to the experiment in question and have no real grounding in dictionaries.

4. Conclusion

It will probably be clear to the reader by now that a crucial problem in assessing this area of work is the fluctuation of the notion of word sense in it, and that is a real problem outside the scope of this paper. For example, sense as between binary oppositions of words is probably not the same as what the Roget categories discriminate, or words in French and English in aligned Hansard sentences have in common.

Another question arises here about the future development of large-scale sense-tagging: Yarowsky contrasts his work with that of efforts like [Cowie et al. 1991] that were dictionary based, as opposed to (unannotated) corpus based like his own. But a difference he does not bring out is that the Cowie et al. work, when optimized with simulated annealing, did go through substantial sentences, mini-texts if you will, and sense-tag all the words in them against LDOCE at about the 80% level. It is not clear that doing that is less useful than procedures like Yarowsky's that achieve higher levels of sense-tagging but only for carefully selected pairs of words, whose sense-distinctions are not clearly dictionary based, and which would require enormous prior computations to set up ad hoc sense oppositions for a useful number of words.

These are still early days, and the techniques now in play have probably not yet been combined or otherwise optimised to give the best results. It may not be necessary yet to oppose, as one now standardly does in MT, large-scale, less accurate, methods, though useful, with other higher-performance methods that cannot be used for practical applications. That the field of sense-tagging is still open to further development follows if one accepts the aim of this paper which is to attack two claims, both of which are widely believed, though not at once: that sense-tagging of corpora cannot be done, and that it has been solved. As many will remember, MT lived with both these, ultimately misleading, claims for many years.

References

Antal, L. "Questions of Meaning". Mouton: The Hague.1963.

Brown, P.F., Di Pietra, S.A., Di Pietra, V.J. and Mercer, R.L. "Word sense disambiguation using statistical methods", *Proc. ACL-91*, 1991.

Bruce, R. and Wiebe, J. Word-sense disambiguation using decomposable models, *Proc. ACL-94*, 1994.

Copestake, A. and Briscoe, "T.Lexical operations in a unification-based framework", *Proc. ACL Siglex Workshop*, Berkeley, 1991.

Cowie, J., Guthrie, J. and Guthrie, "L Lexical Disambiguation using Simulated Annealing", *Proc. Coling-92*, 1992.

Dagon, I. and Itai, A. "Word sense disambiguation using a second language monolingual corpus", *Computational Linguistics*, vol. 20,1994.

Gale, W., Church, K. and Yarowsky, D. "One sense per discourse", *Proc. 4th DARPA Speech and Natural Language Workshop*,1992.

Givon, T. "Transformations of Ellipsis, Sense Development and Rules of Lexical Direction. SP-2896", *Systems Development Corp.*, Sta. Monica, CA., 1967.

Green, G. "Pragmatics and Natural Language Understanding". Erlbaum:
Hillsdale, NJ., 1989.

Hanks, P. personal communication, 1994.

Ide, N. and Veronis, J., "Have we wasted our time?" *Proc. International Workshop on the Future of the Dictionary*, Grenoble, 1994.

Jorgensen, J., "The psychological reality of word senses", *Journal of Psycholinguistic Research*, vol 19, 1990.

Kilgarriff, A., "Dictionary word-sense distinctions: an enquiry into their nature", *Computers and the Humanities*, vol. 26, 1993.

Miller, G. WordNet: a Dictionary Browser, In Proc. First Internat. Conf. on Information in Data. Waterloo OED Centre, Canada, 1985.

Pustejovsky, J., "The Generative Lexicon", *MIT Press*: Cambridge, MA., 1995.

Schuetze, H. and Pederson, J., "Information Retrieval based on Word Sense", *Proc. Fourth Annual Symposium on Document Analysis and Information Retrieval.* Las Vegas, NV., 1995.

Small, S., Cottrell, G., and Tanenhaus, M. (Eds.) "Lexical Ambiguity Resolution", Morgan Kaufmann: San Mateo, CA., 1988.

Wierzbicka, A., "Semantics Culture and Cognition", *OUP: Oxford*, 1989.

Wilks, Y. Grammar, "Meaning and the Machine Analysis of Language", *Routledge: London*, 1972.

Wilks, Y., "Making Preferences more Active". *Artificial Intelligence*, vol. 11, 1978.

Wilks, Y., Fass, D., Guo, C.M., McDonald, J., Plate, T., and Slator, B., "Providing machine-tractable dictionary tools". *Journal of Machine Translation*, vol 5, 1990.

Wilks, Y., Slator, B. and Guthrie, L., " Electric Words", *MIT Press*: Cambridge, MA., 1996.

Yarowsky, D., " Word-sense disambiguation using statistical models of Roget's categories, trained on very large corpora", *Proc. Coling-92*, 1991.

Yarowsky, D., "One sense per collocation", *Proc. ARPA Human Language Technology Workshop*, Princeton, 1993.

Yarowsky, D., "Unsupervised word-sense disambiguation rivalling supervised methods", *Proc.*

ACL-95, 1995.

Acknowledgement

Work referred to was supported by the NSF under grant #IRI 9101232. The paper is also indebted to comments and criticisms from Adam Kilgarriff, David Yarowsky, Karen Sparck Jones, Rebecca Bruce and members of the CRL-New Mexico and University of Sheffield NLP groups. The mistakes are all my own, as always.

Computational Linguistics and Chinese Language Processing
vol. 3, no. 2, August 1998, pp. 17-60
©Computational Linguistics Society of R.O.C.

Information Extraction: Beyond Document Retrieval

Robert Gaizauskas* and Yorick Wilks*

Abstract

In this paper we give a synoptic view of the growth text processing technology of information extraction (IE) whose function is to extract information about a pre-specified set of entities, relations or events from natural language textsand to record this information in structured representations called templates. Here we describe the nature of the IE task, review the history of the area from its origins in AI work in the 1960's and 70's till the present, discuss the techniques being used to carry out the task, describe application areas where IE systems are or are about to be at work, and conclude with a discussion of the challenges facing the area. What emerges is a picture of an exciting new text processing technology with a host of new applications, both on its own and in conjunction with other technologies, such as information retrieval, machine translation and data mining.

1. Introduction: IE and IR

Information extraction (IE) is a term which has come to be applied to the activityof automatically extracting pre-specified sorts of information from short, natural language texts -- typically, but by no means exclusively, newswire articles. For instance, one might scan business newswire texts for announcements of management succession events (retirements, appointments, promotions, etc.), extract the names of the participating companies and individuals, the post involved, the vacancy reason, and so on. Put another way, IE may be seen as the activity of populating a structured information source (or database) from an unstructured, or free text, information source. This structured database is then used for some other purpose: for searching or analysis using conventional database queries or data-mining techniques; for generating a summary; for constructing indices into the source texts.

Information extraction should not be confused with the more mature Technology of information retrieval (IR), which given a user query selects a (hopefully) relevant subset of documents from a larger set. The user then browses the selected documents in order to

* Department of Computer Science, University of Sheffield. E-mail: {robertg, yorick}@dcs.shef.ac.uk

fulfil his or her information need. Depending on the IR system, the user may be further assisted by the selected documents being relevance ranked or having search terms highlighted in the text to facilitate identifying passages of particular interest.

The contrast between the aims of IE and IR systems can be summed up as: IR retrieves relevant documents from collections, IE extracts relevant information from documents. The two techniques are therefore complementary, and their use incombination has the potential to create powerful new tools in text processing.

The differences and complementarity of the techniques can be illustrated by means of an example. The management succession event scenario outlined above was part of the DARPA MUC-6 information system evaluation (see section 2.2.4below). For this evaluation texts pertaining to management succession were required. To obtain them, a corpus of Wall Street journal articles was searched using an IR system (*eg* (5)) with the query shown in Figure 1a). The query was deliberately *not* fine-tuned, as it was desired to obtain some proportion of irrelevant texts. A sample of a relevant text retrieved by this query is shown in Figure 1b). Such texts were then run through IE systems one of whose principal tasks was to fill in a template whose structure is shown in Figure 1c) to produce results as (partially) shown in 1d); as secondary output the system used here is able to generate a natural language summary of the information in the template as shown in e).

Not only do IE and IR differ in their aims, they differ in the techniques they employ. These differences arise partly from their difference in aim, but also for historical reasons. Most work in IE has emerged from research into rule-based systems in computational linguistics and natural language processing, while IR work, where it has not been *sui generis* has been influenced by information theory, probability theory, and statistics. Because of the requirement to extract information, IE must pay attention to the structural or syntagmatic properties of texts: `Carnegiehired Mellon' is not the same as `Mellon hired Carnegie' which differs again from `Mellon was hired by Carnegie'. The simplest IR systems treat texts as no more than `bags' of unordered words. More refined systems allow phrasal matching, proximity searching, and possibly thesaural expansion of query terms. But these techniques are still not adequate to extract, for example, role players in events and their attributes, as the following example shows:

1. 'BNC Holdings Inc. named Ms G. Torretta to succeed Mr. N. Andrews as its new chair-person';

2. 'Nicholas Andrews was succeeded by Gina Torretta as chair-person of BNC Holdings Inc.';

3. 'Ms Gina Torretta took the helm at BNC Holdings Inc. She succeeds

Nick Andrews'.

To extract a canonicalised fact such as `G. Torretta succeeds N. Andrews as chair-person of BNC Holdings Inc.' from each of these alternative formulations, some level of linguistic analysis is necessary -- to cope with grammatical variation (active/passive), lexical variation (`named to' *vs.* `took the helm'), and cross-sentence phenomena such as anaphora.

The inadequacies of IR techniques for getting at the content of texts, and hence their limitations in satisfying text users information needs, have been long known; indeed almost every paper on IE starts with a cry that IR is inadequate (5;5;5). But is progress in IE being made? Are usable systems emerging, or is there a hope that they shortly will? Our aim in writing this paper is to give positive answers to these questions. In section 2 we review the history of IE, giving, if not an exhaustive review, at least a broad feeling for the work that hasgone on in the area. In section 3 we try to give some flavour for the techniques and approaches that have been and are being used in IE systems, concentrating, excusably we trust, on the IE system we have developed and are currently using in a number of research projects. Then, in section 4 we discuss application areas and applied systems, where IE systems are actually performing real world tasks. We conclude, in section 5, by discussing some of the challenges facing IE in the future and the boundaries of IE. Overall we hope to give a reasonable picture of the achievements, limitations, and potential of this exciting new text processing technology.

2. A Brief History of Information Extraction

IE as an area of research interest in its own right was first surveyed in (5). Very broadly one can say that the field grew very rapidly from the late 1980's when DARPA, the US defence agency, funded competing research groupsto pursue IE. However, significant work of relevance was carried out before the DARPA initiative, some of it finding its roots in the 1960s. In this section we divide the work on IE into three broad categories: early work on template filling (work carried out or under way before the DARPA programme); work carried out in response to the DARPA MUC programme; and recent work on IE outside the DARPA programme. This division, like any for review purposes, is crude and not too much weight should be placed upon it.

a) chief executive officer had president chairman post succeed name

b)
```
<DOC>
<DOCNO> 940413-0062. </DOCNO>
<HL>     Who's News:   @   Burns Fry Ltd. </HL>
<DD> 04/13/94 </DD>
<SO> WALL STREET JOURNAL (J), PAGE B10 </SO>
<TXT>
<p>
   BURNS FRY Ltd.  (Toronto) -- Donald Wright, 46 years old, was named executive vice president and director of
 fixed income at this brokerage firm.  Mr. Wright resigned as president of Merrill Lynch Canada Inc., a unit of Merrill
 Lynch & Co., to succeed Mark Kassirer, 48, who left Burns Fry last month.  A Merrill Lynch spokeswoman said it
 hasn't named a successor to Mr. Wright, who is expected to begin his new position by the end of the month.
</p>
</TCT>
</DOC>
```

c)
```
<TEMPLATE> :=
          DOC_NR:
          CONTENT:
 <SUCCESSION_ EVENT> :=
          SUCCESSION_ORG:
          POST:
          IN_AND_OUT:
          VACANCY_REASON:
<IN_AND_OUT> :=
          IO_PERSON:
          NEW_STATUS:
          ON_THE_JOB:
          OTHER_ORG:
          REL_OTHER_ORG:
<ORGANIZATION> :=
          ORG_NAME:
          ORG_ALIAS:
          ORG_DESCRIPTOR:
          ORG_TYPE:
          ORG_LOCALE:
          ORG_COUNTRY:
<PERSON-9301190125-6> :=
          PER_NAME:
          PER_ALIAS:
          PER_TITLE:
```

d)
```
<TEMPLATE-9404130062-1> :=
        DOC_NR: "9404130062"
        CONTENT: <SUCCESSION_ EVENT-9404130062-1>
<SUCCESSION_ EVENT-9404130062-1> :=
        SUCCESSION_ORG: <ORGANISATION-9404130062-1>
        POST: "executive vice president"
        IN_AND_OUT:   <IN_AND_OUT-9404130062-1>
                      <IN_AND_OUT-9404130062-2>
        VACANCY_REASON: OTH_UNK
    IN_AND_OUT-9404130062-1> :=
    IO_PERSON: <PERSON-9404130062-2>
    NEW_STATUS: OUT
    ON_THE:JOB: NO
<IN_AND_OUT-9404130062-2> :=
      IO_PERSON: <PERSON-9404130062-1>
      NEW_STATUS: IN
      ON_THE_JOB: NO
      OTHER_ORG: <ORGANIZATION-9404130062-2>
      REL_OTHER_ORG: OUTSIDE_ORG
<ORGANIZATION-9404130062-1> :=
      ORG_NAME:   "Burns Fry Ltd."
      ORG_ALIAS:   "Burns Fry"
      ORG_DESCRIPTOR:   "this brokerage firm"
      ORG_TYPE:   COMPANY
      ORG_LOCALE:   Toronto CITY
      ORG_COUNTRY:   Canada
<ORGANIZATION-9404130062-2> :=
      ORG_NAME:   "Merrill Lynch"
      ORG_ALIAS:   / "Merrill Lynch"
      ORG_DESCRIPTOR:   "a unit of Merrill Lynch & Co."
      ORG_TYPE:   COMPANY
<PERSON-9404130062-1> :=
    PER_NAME:   "Donald Wright"
    PER_ALIAS:   "Wright"
    PER_TITLE:   "Mr."
<PERSON-9404130062-2> :=
    PER_NAME:   "Mark Kassirer"
```

e) BURNS FRY Ltd. Named Donald Wright
as executive vice president.

Donald Wright resigned as president
of Merrill Lynch Canada Inc..

Mark Kassirer left as president of
BURNS FRY Ltd.

Figure 1 *IR and IE: a) an IR query b) a retrieved text c) an empty template d) a fragment of the filled template e) a 'summary' generated from the filled template*

2.1 Early Work on Template Filling

Applied work on filling structured records with information from natural language texts appears to have originated in two long-term, research-oriented natural language processing projects. The Linguistic String Project (5) at New YorkUniversity began in the mid-60's and carried on into the 1980's. While concerned on the research side largely with the development of a large-scale computational grammar of English, the applications of the work were to do with deriving what Sager called information formats, regularised table-like forms which were, effectively, templates. These information formats abstracted away from the profusionof natural language forms and permitted a database to be defined against which `fact retrieval' (as opposed to document retrieval) could be carried out. The applications were in the medical domain and concentrated on radiology reports andhospital discharge summaries. Some limited evaluation was carried out by contrasting the program's behaviour with the results of getting a human clinician tofill in a comparable information format solely on the basis of the information in the discharge summary. One interesting aspect of this work is that the information formats are *not* predefined *a priori* by experts in the field; rather, given a set of texts in a sub-language domain the information formats (the columns or fields in the tables) are induced by using distributional analysis to discover word classes in the domain (e.g. `film shows clouding', `x-rays indicate metastasis', etc. permit the definition of a **TEST | SHOW | MEDICAL FINDING** format). While inducing templates was abandoned through the 1980's and early 90's as simply too difficult,and the use of predefined, tailored templates created by domain experts adopted instead, there is renewed interest in automatically acquired templates (5).

The second long term project of relevance to the formation of IE as an autonomous area of research was the work on language understanding, and in particular on story comprehension, carried out at Yale University by Roger Schank and his colleagues (5;5;5). Central to this work was the notion that stories followed certain stereotypical patterns which Schank referred to as scripts. Knowingthe script, language comprehenders are able to fill in details and make inferential leaps where the information required to make the leap is not present in the text. Thus a corporate merger, or a management succession event, or a doctor-patient examination all have predictable role-players and sub-events and knowing these permits us to make sense of a text describing any instance of such an event. The first attempt to build what might be called an IE system using this approach was made by one of Schank's students, Gerald De Jong, who designed and built a system called **FRUMP** (5). It used what De Jong called ketchy scripts, a simplified version of the detailed scripts Schank had proposed, to process texts directly from a UPI news wire feed. De Jong's system employed sketchy scripts for sixty situations to extract information from news stories in domains ranging from earthquakes to labour strikes. The instantiated scripts were then used to generate summaries of the stories. His approach relied upon an alternation of predictor and substantiator modules which used, respectively, top-down, expectation-driven processing relying on predictions from the script and bottom-up, data-driven processing based on input from the text. This general approach has been adopted,in one way or another, by many IE systems since. De Jong's work is also notable for carrying out a reasonably extensive evaluation: six days of previously unseen news stories were fed in real-time through FRUMP and the results classified as towhether the stories were processed correctly, nearly correctly, wrongly, or were missed.

Following these initial projects, the 1980's saw the first commercial IE systems developed. The first system to be commercially deployed (to the best of our knowledge) was ATRANS, a system for automatic processing of money transfer messages between banks (5). ATRANS adopted the Yale script-style approach to text processing, using script-driven predictions to identify actors (originating customer, originating bank, receiving bank, etc.) in order to fill in a template that was used, after human verification, to initiate automatic money transfers. Soon after, the Carnegie Group developed and deployed a `fact extraction'system for Reuters called JASPER (5). JASPER was designed to skim company press releases on PR Newswire and fill in a template containing information aboutcompany earnings and dividends. These templates were used to produce candidate news stories which were then validated or post-edited by journalists, offering them a significant savings in story preparation time. A final commercial system initiated in this period was the SCISOR system developed by GE for analysis of corporate mergers and acquisitions (5).

Two other academic research projects from this period should be mentioned. The first was a system developed by James Cowie to extract regularised descriptions (effectively, templates) of plants fromwild flower guides (5). Cowie's approach relied upon a domain-specific, handcrafted lexicon of keywords which allowed segments of the source text to be matched with appropriate sectionsof the target template. Rules pertaining to slots in the template (properties of plants) were then brought to bear on the selected portions of text and the propertyvalues extracted. The second was a project by G.P. Zarri to translate automatically French texts dealing with a particular period of French history into a `metalanguage' which captured certain semantic relations pertaining to biographical details that were sought (5). This metalanguage was organised around case frames for predicates, which can be viewed as small-scale templates: what was to be extracted were the roles in particular historical events, such as the naming to a position of an historical figure by a given body on a particular date at some location. The approach involved first using a syntactic analyser to establish the text's syntactic structure, and then carrying out semantic parsing in which lexical triggers -- keywords in the domain -- caused one or more of the case frames for key predicates to be invoked and then instantiated with material identified from thesyntactic analysis, according to rules associated with the slots case frame slots.

2.2 The Message Understanding Conferences - MUC

2.2.1 Background to MUC

In the mid-1980's a number of sites in the US were working on IE from naval messages, in projects sponsored by the US Navy. In order to understand andcompare their systems' behaviour better, a number of these message understanding (MU) projects decided to work on a set of common messages and then convene tosee how their systems would perform when given some new, unseen messages. This gathering constituted the first of what has turned into an ongoing series of extremely productive message understanding conferences, or MUCs, which haveserved as key events in driving the field of IE forward (the term `message under-standing' is now disappearing in favour of the more descriptively accurate `information extraction')(5;5;5;5).

There have been six Message Understanding Conferences to date and a seventh is planned for spring 1998. The objective of the conferences has been to establish a quantitative evaluation regime for IE or MU systems, which prior to these conferences had been sporadically assessed in an *ad hoc* fashion, frequentlyon the same data on which they had been trained. To date, the MUC conferenceshave been sponsored by DARPA and organised by the US Naval Command,Control, and Ocean Surveillance Center RDT\&E Division (NRaD), formerly theNaval Ocean Systems Center, in San Diego, California.

A brief chronology and description of the MUCs is as follows:

MUC-1 Held in May 1987 in San Diego. Six systems participated. The texts were tactical naval operations reports on ship sightings and engagements. Twelve training reports were supplied, plus additional messages. Two unseen messages were distributed at the conference for participants to test their systems on. There was no task definition and there were no evaluation criteria.

MUC-2 Held in May 1989 in San Diego. Eight systems participated. Again the domain was tactical naval operations reports on ship sightings and engagements. 105 messages were supplied as training data and there were two test rounds, one with 20 blind messages and then, after system fixes, a second round of 5 blind messages just before the conference. This time a task was specified: a template was defined and fill rules for the slots supplied. Answer keys, i.e. correctly filled templates, were manually prepared for Development and test texts. Resources in the form of lists of specialised naval terminology were also supplied. Evaluation criteria were defined, but by consensus deemed not to have been adequate. Scoring was done by participating sites.

MUC-3 Held in May 1991 in San Diego. Fifteen systems participated. The domain was newswire stories about terrorist attacks in nine Latin American countries. The stories were gathered from an electronic database but were originally items as diverse as newspaper stories, radio and television broadcasts, speeches, interviews, news conference transcripts, and communiques. Most were translated from Spanish by the US Foreign Broadcast Information Service. 1,300 development texts were supplied and three blind test sets of 100 texts each were prepared. A template was defined consisting of 18 slots. Formal evaluation criteria were introduced, adapted from notions developed in information retrieval (specifically, precision and recall). A semi-automated scoring program was developed and made available for use by participants during development. Official scoring was done by the organisers.

MUC-4 Held in June 1992 in McLean, Virginia. Seventeen sites participated. The domain (Latin American terrorism) and template structures remained essentially unchanged. Changes were made to the task definition, corpus, measures of performance, and test protocols in order to provide greater focus on spurious data generation, to better assess system independence from training data, to make scoring more consistent, and to provide means for more valid score comparison between systems. This evaluation marked the beginning of the inclusion of the MUC conferences within the TIPSTER text programme[1]

MUC-5 Held in August 1993 in Baltimore, Maryland (coinciding with the TIPSTER-I 24-month evaluation). Seventeen systems participated (fourteen American, one British, one Canadian and one Japanese -- this marked the first non-US involvement). Two domains -- joint ventures in financial newswire stories and microelectronics products announcements -- and two languages -- English and Japanese -- were tested. Substantial ancillary resources were supplied. Development and test corpora sizes were increased. Scoring was modified to include new evaluation metrics and the scoring program enhanced. More details of MUC-5 are presented in Section 2.2.3.

[1] TIPSTER is a U.S. Government programme of research and development in the areas of IR and IE.

MUC-6 Held in November 1995 in Columbus, Maryland. Seventeen sites overall took part. The evaluation emphasized finer-grained evaluation and portability issues and comprised four subtasks -- named entity recognition, coreference identification, and template element and scenario template extraction tasks. The domain of the scenario extraction task was management succession events in financial news stories. Sites were allowed to choose which subtasks they would undertake. MUC-6 is discussed further in section 2.2.4 below.

Across these evaluation exercises, the tasks have become progressively more difficult. Some effort was made to quantify this increase at MUC-5 and the conclusion drawn that there was an order-of-magnitude increase in task complexity on several measures between MUC-2 and MUC-5 (5). Task complexity measures included text corpus complexity (e.g. vocabulary size, average sentence length), textcorpus dimensions (e.g. volume of texts, total number of sentences/words), templatecharacteristics (e.g. number of object types, number of slots), and difficulty of task (hard to measure, but considered, e.g., number of pages of relevance rules and template fill definitions). System performance has improved against this backdrop of increasing task complexity, indicating that genuine progress in developing this technology has been made in the past decade.

In sections 2.2.3 and 2.2.4 we describe MUC-5 and MUC-6 in some detail, as the most recent and most sophisticated IE evaluations.

2.2.2 Evaluation metrics

The evaluation metrics have evolved with each MUC. The starting points for the development of these metrics were the standard IR metrics of recall and precision. In the information extraction task, recall may be crudely interpreted as a measure of the fraction of the required information that has been correctly extracted and precision as a measure of the fraction of the extracted information that is correct. The definitions of these measures have been altered from those used in IR (but the names have been retained) to allow for overgeneration in IE where, unlike IR,data not present in the input can be erroneously produced.

Not only have recall and precision measures been redefined for the extraction task, but additional measures have been introduced as well. Slot fills can be correct, partially correct, or incorrect, but they can also be missing (no fill when there should be), sprious

TIPSTER is not an acronym and appears to have been adopted as a name because of the intelligence providing potential of these technologies (*cf*. the Oxford Concise Dictionary: **tipster** *n*. a person who gives tips, esp. about betting at horse-races.)

(fill present when it should not be), or non-committal (no fill when the answer key also contains no fill). These extra categories permit the introduction of measures of overgeneration (fraction of extracted information that is spurious), undergeneration (fraction of information to have been extracted that is missing), and substitution (fraction of the nonspurious extracted information that is not correct).

For MUC-3 and MUC-4 recall and precision were the primary metrics and the others were secondary. In addition, for MUC-4, van Rijsbergen's combined measure of recall and precision, the F-measure, was used (5). But for MUC-5, recall and precision were deemed unofficial metrics and a new primary metric called error per response fill was introduced. This was an attempt to measure the fraction of a system's response that is `wrong', i.e. the fraction of the combined actual and possible responses that were faulty. It was hoped that this measure would allow developers to focus more directly on the sources of their systems' difficulties, in particular on missing and spurious information which figures directly in the error-based metric, but only indirectly in the recall and precision metrics. In MUC-6 recall and precision regained their status as official metrics and the metrics were slightly modified so as to eliminate the category of partially correct slot fill. All of these metrics carried over to three of the four MUC-6 tasks, but only precision and recall metrics were employed for the coreference task and their definitions had to be modified to account for peculiarities of this task (see (5) for more details).

Since at least MUC-3, a text-filtering metric has also been employed to measure how good systems are at separating documents into relevant/nonrelevantcategories. This measure operates at the level of texts as a whole (are templates generated for a given text when they should be or not) and not at the level of slots.

2.2.3 MUC-5

Task As with MUC-3 and MUC-4, the MUC-5/TIPSTER-I 24-month evaluationrequired systems to extract information from newswire stories. There were four possible tasks: two domains (joint ventures and microelectronics) and two languages(Japanese and English). These domain-language pairs are referred to using the acronyms EJV, JJV, EME and JME, in the obvious way. Participating non-TIPSTER-sponsored systems had to choose one domain and either or both languages; TIPSTER-sponsored systems were intended to operate in all four domain/language pairs. Most sites did only one task as this proved more than challengingenough. The EJV task was the most popular, and by common consent the most difficult; most of the following detailed remarks pertain to this task.

The MUC-5 template and fill rules were the most complex to date. For the first time the template was not a flat data structure, but rather allowed slots to contain pointers to other slots. Thus the template had an `object-oriented' feel. For example, a joint venture was viewed as an object with various slots including its name and status (`existing', `dissolved', etc), but also slots for the participating organisations, each of which was to be filled with a pointer to an organisation object, itself containing slots which in some cases contained pointers to other complex objects. In all there were 11 objects and 49 slots to be filled in. Slotswere of four types: set fills (contained one of a given set of alternatives -- e.g. organisation type could be company, person, government or other); string fills (contained a copy of some string from the original text -- e.g. company name); normalised entries (contained data from the text transformed into a canonical form -- e.g. dates, times, monetary amounts); references (pointers to other objects, as described above). As an indication of the level of detail required to define the extraction task, the fill rules occupied a 45 page document.

Resources There were three sources for the EJV materials: the Wall Street Journal, Lexus/Nexus, and PROMT. Roughly 2300 training texts were provided andanswer keys were supplied for most of them. There was a dry run blind test set of 200 articles provided roughly half way through the evaluation, and a final blindtest set of 286 articles. Official scoring was done for both dry run and final tests by MUC organisers but the scoring program was made available to all sites for use during development. This program was an extremely sophisticated piece of software which could be run in an entirely automatic mode, or in an interactive modewhere the scorer is queried about the status of what the program judges may be partially correct answers.

The texts ranged in length from just two or three sentences, to several pages.Sentence lengths varied enormously, but some of length greater than seventy wordswere reported. In some places the texts contained tabular numeric data. The texts varied between mixed case and all upper case. All were originally marked up in SGML and contained certain reliably extractable information such as document id, date and source, flagged by SGML markers.

In addition to the training corpora and answer keys, considerable other data resources were supplied. These included: gazetteer of place names (246,908 entries);list of corporate names and nationalities (50,759 entries); list of corporate designators (133 entries); list of countries (244 entries); list of nationalities (216 entries); list of international organisations (~175 entries); definitions of (American) standard industry codes (17,779 entries); list of currency names/nationalities (217 entries); list of female forenames (4967 entries); list of male forenames (2924 entries); CIA world fact book.

Some of the previous participants also made utilitysoftware available.

The methodology and effort required to produce the answer keys were both-nontrivial. The production of the templates was undertaken by a small team of analysts, equipped with workstations and a software tool to aid in the extraction task. An elaborate procedure of selecting subsets of the documents to be multiply analysed was adopted in an attempt to ensure consistency in the answer keys. Of course the fill rules had to be modified as new complexity was uncovered and thisrequired correcting previously created answer keys. The cost of producing the answer keys alone for MUC-5 and for the preceding TIPSTER extraction trials wasmore than $1 million US.

Results Table 1 shows the best raw score obtained in each of the four tasks discussed above. One interesting thing to note from these results is that in each domain the Japanese scores were higher. This observation has prompted discussion of whether in some sense Japanese is an easier language from which to extract information.

For error per response fill, undergeneration, overgeneration, and substitution the lower the score the better; for recall and precision the higher the score the better. Raw scores need to be interpreted very cautiously. Statistical studies were done on them (5) and for each task a number of ranks were identified within which raw score differences were claimed to be of no significance. For EJV there were 7 statistically significant ranks into which 13 systems were placed; in JJV 3 ranks for 5 systems; in EME 5 ranks for 7 systems; and in JME 2 ranks for 4 systems.

2.2.4 MUC-6

Tasks In MUC-6, rather than a single `end-to-end' system evaluation as in MUC-5, participants were offered a menu of smaller evaluations from which they could pick and choose, depending on their interests and available resources. There were four evaluated tasks.

Task	ERR	UND	OVG	SUB	REC	PRE	P & R
EJV	61	30	39	19	57	49	52.8
JJV	50	32	23	12	60	68	63.8
EME	65	37	41	19	50	48	49.2
JME	58	30	38	14	60	53	56.3

Table 1. *MUC-5 Best Overall Raw Scores indicating error per response fill (ERR), undergeneration (UND), overgeneration (OVG) substitution (SUB), recall (REC), precision (PRE) and combined precision and recall (P & R / F-measures) (from (5))*

1. Named entity recognition. This task required the recognition and classification of definite named entities such as organisations, persons, locations, dates and monetary amounts. Classes of entity were reported by marking up the source text with SGML. In the usual MUC fashion, scoring involved comparing the system's proposed result with manually prepared answer keys. Here is a simple example:

<enamex type="organization">Bridgestone Sports Co.</enamex> said
<timex type="date">Friday</timex> it has set up a joint venture in
<enamex type="location">Taiwan</enamex> with a local concern and a
Japanese trading house to produce golf clubs to be shipped to
<pnamex>Japan</pnamex>.

where enamex indicate an entity name, timex a time expression, and pnamex a place name expression.

2. Coreference resolution. This task required the identification of expressions in the text that referred to the same object, set or activity.

Once again SGML markup was used to annotate coreferential expressions. For example

<coref id="100">Galactic Enterprises</coref> said<coref id="101" type="ident"
ref="100">it</coref> would build a new space station before the year 2016.

The id attribute serves to identify arbitrarily, but uniquely, each string taking part in a coreference relation. The **ref** attribute indicates which string is coreferential with the one which it tags. The **type** attribute serves to indicate the relationship between anaphor and antecedent. The value ident for this attribute indicates identity, and in the final MUC-6 task definition was the only relationship to be marked. Other relationships such as **part-whole** and **set-member** had been considered, but were omitted due to difficulties in defining the task precisely enough.

Coreference relations were only marked between certain syntactic classes of expressions (noun phrases and pronouns) and a relatively constrained class of relationships to mark was specified, with clarifications provided with respect to bound anaphors, apposition, predicate nominals, types and tokens, functions and function values, and metonymy.

3. Template element filling. This task required the filling of small scale templates wherever they occurred in the texts. There were only two such template elements, one for organisations and one for persons. These are illustrated in Figure 1.

4. Scenario template filling. The task required the detection of specific relations holding between template elements relevant to a particular information need (in this case corporate management personnel joining and leaving companies) andconstruction of an object-oriented structure recording the entities and details of the relation. This is illustrated in Figure 1.

The precise specifications of each of these tasks may be found in Appendices C-Fof (5).

Four other evaluations had been considered, but were dropped due to lack of agreement over task definitions and lack of time and money for producing the development and test resources. These were parse structure evaluation (provide a canonical syntactic analysis of each sentence); predicate-argument structure evaluation (provide a canonical semantic analysis of each sentence); word sense disambiguation (disambiguate the sense of each open class, non-proper name word with respect to some standard lexical resource such as WordNet (5)); and cross-document coreference (determine coreferences between distinct documents).

The demand for this restructuring of the evaluation exercise arose for a number of reasons. Different participants had different interests and believed effort should be focussed in different areas. End-to-end systems IE were getting bigger and bigger and many research groups were excluded simply because they could notput the resources together to produce a massive system, where software engineeringissues can soon come to eclipse research issues. Furthermore, comparison of systems andapproaches had proved extremely difficult because the grain of the evaluation was too large. Finer scale evaluation, it was believed, would focus and promote more fruitful debate. However, it can be argued that any subdivision of the end-to-end IE task presupposes a processing approach to the task which may inhibit radically new approaches from emerging.

Resources As with MUC-5, the principal resources supplied by the organisers were annotated development and test corpora and scoring software. For both the dry run and final evaluations, 100 annotated development texts were provided for each of the four tasks. For the evaluations themselves there were 30 annotated test texts for the named entity and coreference tasks, and 100 annotated test texts for the scenario template and template element tasks. These texts were all WallStreet Journal texts, all of them mixed case. New scoring software was developed for the named entity and coreference tasks, and the MUC-5 scoring software enhanced for the template tasks.

Evaluation In MUC-6 the official evaluation metric reverted to precision and recall from the error-per-response-fill metric used in MUC-5. These two metrics had shown themselves to be very closely in line in MUC-5 and participants generally preferred precision and recall (perhaps because one tries to maximise these measures, whereas one tries to minimise error-per-response-fill, which caststhe whole exercise in a more negative light).

The two template filling tasks were scored as in previous MUCs, with improvements to the scoring software, but no major departures. The named entitytask required a new scorer based on comparing SGML-marked up strings, but the standard definitions of recall and precision carry over quite naturally here. However, in the coreference task, a problem arises which requires that the precision and recall scoring measures be specially adapted. Clearly, more than twomarkables may corefer, i.e., there may be chains of coreferences, not simply coreferential pairs. In the case of chains, how to record the chain and how to score systems which fail to discover all the links in the chain become central issues. See (5) for a full discussion of the definitions of precision and recall for the coreference task.

Task	ERR	UND	OVG	SUB	REC	PRE	P & R
Named Entity	5	2	1	2	96	97	96.42
Coreference (High Recall)					63	63	
Coreference (High Precision)					59	72	
Template Element	29	20	5	8	74	87	79.99
Scenario Template	57	41	12	20	47	70	56.40

Table 2. *MUC-6 Best overall Raw Scores indicating error per response fill (ERR), undergeneration (UNG), overgeneration (OVG) substitution (SUB), recall (REC), precision (PRE) and combined precision and recall (P & R / F-measure) (from (5))*

Results Table 2 shows the best raw score obtained in each of the four tasks. In all but the coreference case the results of the system with the best combined precision and recall score (F-measure) have been displayed (thus, there may be other systems which obtained higher scores on one of the other measures). Due to differences in the approach to scoring the coreference task and the other tasks, only recall and precision measures were available for coreference, and no satisfactory combined measure could be defined.

2.2.5 An Assessment of MUC

Even after doing statistical significance studies it is hard to come to any firm conclusion about the superiority of a given approach, principally because of the varying levels of resources that different sites brought to the task -- person-months spent on development, qualifications and backgrounds of the people doing the development, software and hardware resources committed, and so on. At the conference every site could put up a graph showing a steep line of improvement from the immediately preceding dry run evaluation and claim (especially to their funding bodies !) that given another few months they could make spectacular gains. Clearly this improvement has to stop somewhere; but there is no way of telling which approach will level out when and at what level.

Another criticism frequently made of the MUC evaluations is that they lead to copy-cat behaviour, whereby systems tend to converge upon the same approach because any advantage is quickly picked up by others afraid to lag behind in the short term because of funding implications of being seen to be a `loser'.

Each of these criticisms can be at least partially answered. The first one -- that the evaluation results do not let us draw unequivocal conclusions -- by observing that imperfect evaluation is better than none at all. The results can tell us important things; we simply need to be careful in interpreting the results. The second criticism -- that participating sites tend to play safe by copying successfulapproaches -- may be true of some sites (perhaps those directly dependent on linked funding), but is certainly not true of all sites, particularly academic ones (section 3.3.1 gives some indication of the wide range of approaches still being entertained). Besides the rapid transfer of successful technology can hardly be viewed as completely deleterious.

In all the MUC evaluations have provided the IE community resources,evaluation tools, and perhaps above all a sense of identity and a forum for exchange of ideas. There may come a time when their utility becomes questionable; but they have proved of significant worth to date.

2.3 Other Work on Information Extraction

The MUC evaluations are still running, but concurrent with them, either unrelatedlyor in part because of the higher interest in IE they have generated, numerous otherIE projects can be identified. This list describes some significant European IE projects, but it is almost certainly incomplete given the rapidly expanding nature ofthe field.

Two projects which started in the late 1980's illustrate the use IE systems forprocessing sublanguages -- specialised languages that are developed within a restricted area of human activity and which are frequently characterised by extragrammaticality (from the perspective of the `mother' language), idiosyncratic lexical forms, and heavy use of ellipsis (because of the shared world knowledge which the context which gives rise to the sublanguage supplies). The first of theseis the POETIC (Portable Extendable Traffic Information Collator) system (5) whosefunction was to extract information about road traffic incidents causing traffic congestion from police incident logs and to generate advisory bulletins to be broadcast to motorists. Police incident logs form a sublanguage in the sense defined above, and the system utilised a special grammar and lexicon, as well as a domain-specific reasoning component to deal with the highly telegraphic and idiosyncratic forms found in the police logs.

The second system was SINTESI (Sistems INtegrato per TESti in Italiano) which processed short texts describing car faults and filled in a template identifying the main fault, chain of causes, chain of effects, car parts involved etc.(5). Once again, because of the nature of the sublanguage, the approach relied extensively on domain-specific lexical-semantic knowledge (caseframes for relevant objects in the domain).

The Language Engineering (LE) initiatives within the Commission of the European Communities (CEC) Third and Fourth Framework programmes have supported a number of IE projects, several of which are currently underway. Theseare simply listed with references for the interested reader, as there is not space to describe them, and in some cases, as the projects are just underway, there is yet little published material about them. The TREE (TRans European Employment) project aims to make information available to job seekers across the European Union by extracting job details from electronic job advertisements and storing them in a database which can be browsed by job seekers in their own language (5;5). The FACILE (Fast Accurate Categorisation of Information using Language Engineering) project, following on from the COBALT project aims to categorise and filter news stories of interest to stock market traders, using extraction-like techniques (5;5;5). Finally, at Sheffield we are working on two applications of IE systems within the CEC LE projects: one, AVENTINUS is in the classic IE tradition, seeking information on individuals about security, drugs and crime, andusing classic templates (5;5). The other, ECRAN, a more research-orientated project, searches movie and financial databases and exploits the notion we mentioned of tuning a lexicon so as to have the right contents, senses and so on to deal with new domains and relations unseen before (5).

3. Approaches to Information Extraction

Since IE systems are large, complex software systems usually consisting of many components, classifying them is not an easy task. Perhaps the most useful aid in this task is a description of the generic IE system provided by J. Hobbs (5). His description allows newcomers to the field to grasp the principal processing stagesinvolved in IE and provides IE system developers with a standard system description against which to differentiate their own. While this description was derived as a synthesis of the approaches used in MUC-4 systems, it remains broadly true.

Armed with this general description we then turn to a description of the LaSIE

(Large Scale Information Extraction) system which we have developed at Sheffield, using the system we know best to illustrate in more detail the sorts of processing involved in information extraction. While LaSIE is quite distinct frommany IE systems, it is not difficult to see how it fits Hobbs's general rubric. Following this moderately detailed description of how one IE system works, we conclude this section with a discussion of some of the general trends that are currently influencing the direction of IE system development.

3.1 The Generic IE System

Hobbs describes the generic IE system as a ``cascade of transducers or modules that at each step add structure and often lose information, hopefully irrelevant, by applying rules that are acquired manually and/or automatically" ((5), p. 87). To describe such a system requires identifying the modules, identifying each module's input and output, identifying the form of the rules the modules apply, and specifying how the rules are applied and how they are acquired.

According to Hobbs, a typical IE system consists of a sequence of ten modules:

1. Text Zoner. Divides the input text into a set of segments.
2. Preprocessor. Converts a text segment into a sequence of sentences, where each sentence is a sequence of lexical items, with associated lexical attributes (e.g. p art-of-speech).
3. Filter. Eliminates some of the sentences from the previous stage by filtering out irrelevant ones.
4. Preparser. Detects reliable small-scale structures in sequences of lexical items (e.g. noun groups, verb groups, appositions).
5. Parser. Analyses a sequence of lexical items and small-scale structures and attempts to produce a set of parse tree fragments, possibly complete, which describes the structure of the sentence.
6. Fragment Combiner. Turns a set of parse tree or logical form fragments into a parse tree or logical form for the whole sentence.
7. Semantic Interpreter. Generates a semantic structure or meaning representation or logical form from a parse tree or parse tree fragments.
8. Lexical Disambiguation. Disambiguates any ambiguous predicates in the logical form.

9. Coreference resolution or discourse processing. Builds a connected representation of the text by linking different descriptions of the same entity in different parts of the text.

10. Template generator. Generates final templates from the semantic representation of the text.

Of course not all systems exhibit all of these modules, nor do they necessarily perform their processing in exactly this sequence (in particular stages 6 and 7 may occur in the reverse order).

3.2 LaSIE: A Case Study

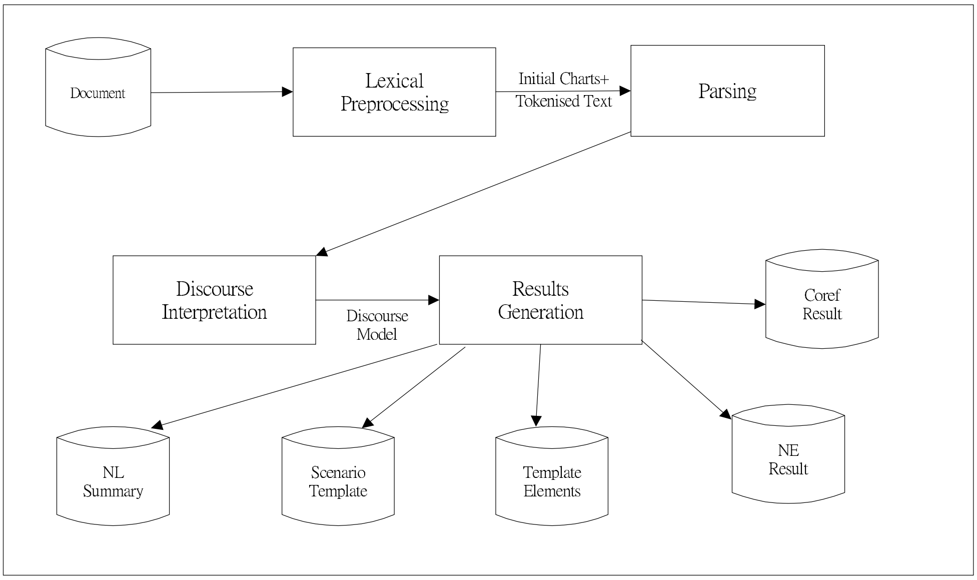


Figure 2
LaSIE System Architecture

LaSIE was designed as a general purpose IE research system, initially geared towards, but not solely restricted to, carrying out the tasks specified in MUC-6: named entity recognition, coreference resolution, template element filling, and scenario template filling. In addition, the system can generate a brief natural language summary of any scenario it has detected in the text. All of these tasks are carried out by building a single rich model of the text -- the discourse model -- from which the various results are read off.

The high level structure of LaSIE is illustrated in Figure2. The system is a pipelined architecture which processes a text one sentence at a time and consists of three principal processing stages: lexical preprocessing, parsing plus semantic interpretation, and discourse interpretation. The overall contributions of these stagesmay be briefly described as follows:

lexical preprocessing reads and tokenises the raw input text, tags the tokens with parts-of-speech, performs morphological analysis, performs phrasal matching against lists of proper names;

parsing and semantic interpretation builds lexical and phrasal chart edges in a feature-based formalism then does two pass chart parsing, pass one with a special named entity grammar, pass two with a general grammar, and, after selecting a `best parse', constructs a predicate-argument representation of the current sentence;

discourse interpretation adds the information from the predicate-argument representation to a hierarchically structured semantic net which encodes the system's world model, adds additional information presupposed by the input, performs coreference resolution between new and existing instances in the world model, and adds any information consequent upon the new input.

Subsequent to MUC-6, LaSIE was re-engineering at the architectural level to make it function within a language engineering research architecture called GATE -- the General Architecture for Text Engineering also developed at Sheffield. GATE is a software environment that supports researchers who are working in natural languageprocessing and computational linguistics and developers who are producing and delivering language engineering systems (5;5). It is based on the TIPSTER architecture (5), an object-oriented data model designed to support a broad range ofdocument processing tasks and promoted as a standard for the information retrievaland extraction tasks within the DARPA-sponsored TIPSTER text programme. The re-engineered LaSIE system functioning within GATE is called VIE (Vanilla IE system). It was derived from LaSIE by standardising LaSIE module interfaces so that all modules communicated with each other via the GATE document manager (allowing for easy substitution of improved modules with similar functionality -- e.g., better part-of-speech taggers, or parsers). Further details of LaSIE and VIE can be found in (5;5).[2]

The processing of the system is best illustrated by means of an example. We will discuss what processing goes on each of the three principal stages identified above with respect to the small text shown in Figure 1b).

[2] GATE and VIE are both publicly available: see http://www.dcs.shef.ac.uk/research/group/nlp/gate for details.

3.2.1 LaSIE: Lexical Processing

This stage comprises five modules.

1. Tokenisation. This module does both text segmentation and tokenisation. In the example text it distinguishes the document header (everything preceding the **<TXT>** tag) from the document body, and in longer texts would segment the text into paragraphs. Tokenisation involves identifying which sequences of characters will be treated as individual tokens -- for example, treating **SGML** tags as single tokens, but separating other punctuation from preceding characters (so **<TXT>** is a token but **Ltd**., in the first line of the text is three tokens).

2. Sentence splitting. This module determines sentence boundaries in the text -- a non-trivial task as full stops are not sufficient guides. For example, they may occur in names (**Allan J. Smith**) and after abbreviations (**Inc. Mr**.), though of course the latter may end sentences too".

3. Part-of-speech tagging. We have used a modified version of the rule based part-of-speech tagger developed by E. Brill (5). It processes one sentence (sequence of tokens) at a time and associates with each token one of the forty-eight part-of-speech tags in the University of Pennsylvania tagset (5). Thus, for input such as Donald Wright, 46 years old the tagger produces output of the form **Donald/NNP Wright/NNP ,/COMMA 46/CD years/NNS old/JJ**, where **NNP** designates a proper noun, CD a cardinal number, **NNS** a plural common noun, and **JJ** an adjective.

4. Morphological analysis. This module does a limited form of morphological analysis, determining root forms of nouns and verbs. In our example **years** will analysed as having root **year** and affix **s** and **named** would be analysed as having root **name** and affix **ed**.

5. Gazetteer lookup. We employ 5 gazetteers, or lists of names, to facilitate the process of recognising and classifying named entities. These are organisation names, location names, personal given names, company designators (**Corp., Ltd**., etc.), and personal titles (**Mr., President**), etc. In our example text, **Toronto** and **Canada** are tagged as places, **Donald** and **Mark** as first names, **executive vice president** and **president** as personal titles and **Ltd., Inc**. and **Co.** as company designators. Only well known names are stored in these lists, so, for example, while **Merrill Lynch** and **Burns Fry** are prestored, a company such as **Sheffield Motor Repairs** would not be.

In addition we use four lists of trigger words, to tag words which occur inside multi-word proper names, and which reliably permit the class of the proper name to be determined. For example, `Wing and Prayer Airlines' is almost certainly a company, given the presence of the word Airlines; `Bay of Pigs' almost certainly a location given the word Bay. This and further aspects of the system's algorithm for proper name recognition are discussed further in (5).

3.2.2 LaSIE: Parsing

The parsing and semantic interpretation stage of LaSIE is carried out by a single module. However this stage consists of three substages. The first substage is parsing with a special named entity grammar. We use a bottom-up chart parser (5) and a manually constructed context-free grammar of 177 rules pertaining to named entities to recognise multi-word structures which identify organisations, persons, locations, dates, and monetary amounts. For example, a rule like **ORGAN_NP --> ORGAN_NP LOC_NP CDG** allows us to recognise the organisation name **Merrill Lynch Canada Inc.** and a rule like **PERSON_NP --> FIRST_NAME NNP** allows us to recognise the person name Donald Wright. Semantic interpretation is carried out in parallel with parsing. This amounts to assigning a regularised form in a predicate-argument notation to each phrase identified by the grammar. For proper names this logical form consists of two terms, a unary predicate specifying the type of the entity and a binary predicate specifying the actual name string. For example, **Burns Fry Ltd.**, following its syntactic analysis, isassigned the logical form **organization(e17), name(e17,'Burns Fry Ltd.'**) where e17 is a unique new identifier introduced to provide an unambiguous handle for the entity referred to in the text as **Burns Fry Ltd.**.

The second substage is parsing with a more general phrasal grammar. The Same parser mechanism is used, but this time with a grammar of 110 rules Designed to recognise noun phrases, verb phrases, prepositional phrases, adjectival phrases, sentences, and relative clauses. This grammar was extracted from a large manually annotated corpus of newswire text, the Penn Treebank (5), using a set of programs designed for the purpose (5). Again, a semantic interpretation is built up during parsing. For instance the sentence **Donald Wright, 46 years old, was named executive vice president and director of fixed income at this brokerage firm** is parsed and assigned a top level structure as shown in figure 3. Note that this analysis is partial due to lack of coverage in the grammar; however,this does not prevent useful information from being derived. From the structural relations that are identified a logical form may assigned. For key parts of this
sentence this takes the form:

person(e21), name(e21, 'Donald Wright')
name(e22), lobj2(e22,e23)
title(e23,'executive vice president')
firm(e24), det(e24,this)

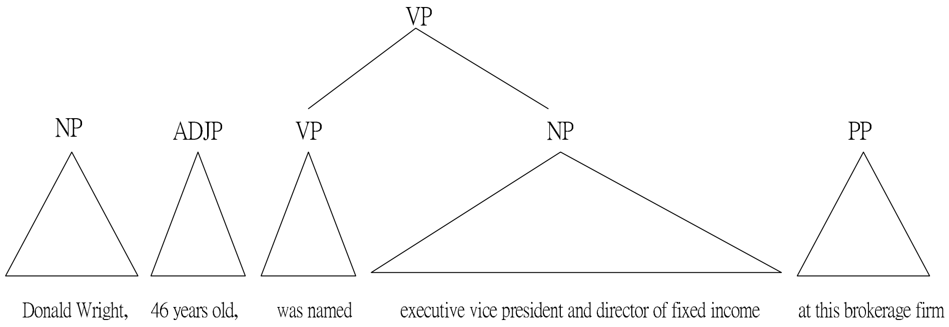


Figure 3 *A LaSIE Parse Forest*

Despite the fact that the parser is complete, *i.e.* finds all structural analyses ofits input sentence according to the grammar, it is rare that these analyses contain aunique, spanning parse of the sentence. Consequently, the final substage of the Parsing module involves selecting a ``best parse'' from the set of partial,fragmentary, and possibly overlapping (and hence incompatible) phrasal analyses which the parser has found. This is currently done by choosing that sequence of non-overlapping phrases of semantically interpretable categories (sentence, noun phrase, verb phrase and prepositional phrase) which covers the most words and consists of the fewest (hence largest) phrases.

3.2.3 LaSIE: Discourse Processing

The principal task of the discourse processing module in LaSIE is to integrate the semantic representations of multiple sentences into a single model of the text from which the information required for filling a template may be derived. The discourse processor works on the semantic representations passed onto it from the parser, though these include a record of the surface text from which they were derived, and in particular permit the order in which entities were introduced to be recovered.

The discourse interpretation stage of LaSIE relies on an underlying `world model', a declarative knowledge base that both contains general conceptual knowledge and serves as a frame upon which a discourse model for a multi-sentence text is built. This world model is expressed in the XI knowledge representation language (5) which allows straightforward definition of cross-classification hierarchies, the association of arbitrary attributes with classes or individuals, and the inheritance of these attributes by individuals.

The world model consists of an ontology plus an associated attribute knowledge base. In LaSIE the ontology consists mostly of classes or `concepts' directly relevant to a specific template filling task. So, for example, for the management succession scenario the ontology is constructed to contain details aboutpersons, posts, and organisations, and also about events involving persons leaving or taking up posts in organisations.

Associated with each node in the ontology is an attribute-value structure. Attributes are simple **attribute:value** pairs where the value may either be fixed, as in the attribute animate:yes which is associated with the person node, or where the value may be dependent on various conditions, the evaluation of which makes reference to other information in the model. Certain special attribute types, **presupposition** and **consequence**, may return values which are used at particular points to modify the current state of the model, as described in the following section. The set of attribute-value structures associated with the whole ontology is referred to as the attribute knowledge base.

The higher levels of the ontology for the MUC-6 management succession extraction task are illustrated in figure 4, along with some very simple attribute-value structures.

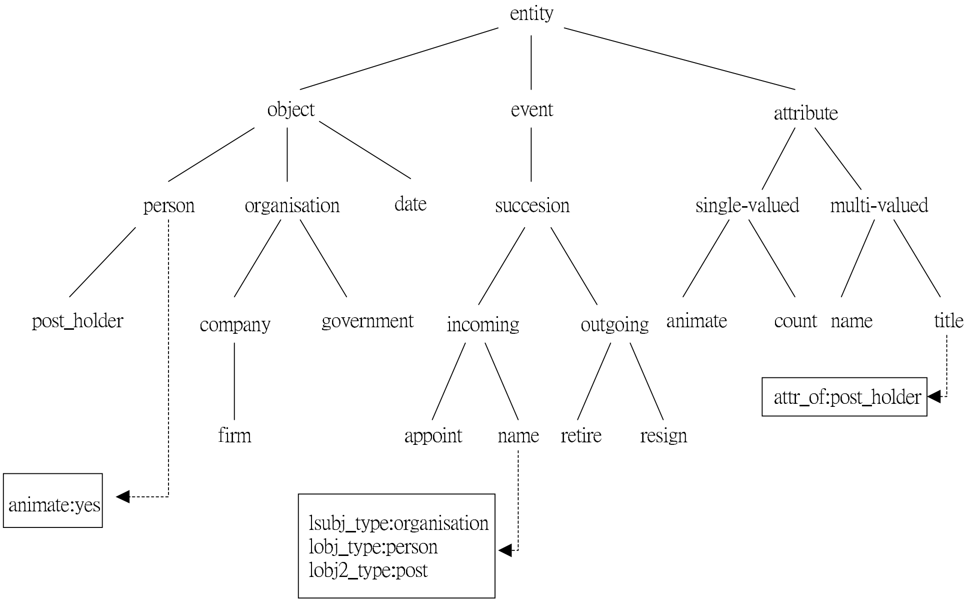


Figure 4 *A Fragment of the LaSIE World Model and Associated Attribute Knowledge Base*

The world model described above can be regarded as an empty shell or frame to which the semantic representation of a particular text is added, populatingit with the instances mentioned in the text. The world model which results is then a model specialised for the world as described by the current text; we refer to this specialised model as the discourse model.

Figure 5 illustrates how instances are added to the world model, specialising it to convey the information supplied in a specific text. In the figure instances are indicated with the notation **e20, 21**, etc. and are shown connected by dashed lines to their classes. The figure reflects the state of discourse processing part way through the interpretation of the sentence `Donald Wright, 46 years old, was namedexecutive vice president and director of fixed income at this brokerage firm', as will be described below. Instances shown in bold derive from previous text (just **e20** in this case, derived from the dateline), instances in normal font indicate entities deriving directly from the current sentence, and those in italic font (just *e25* here) are instances hypothesised in processing the current sentence.

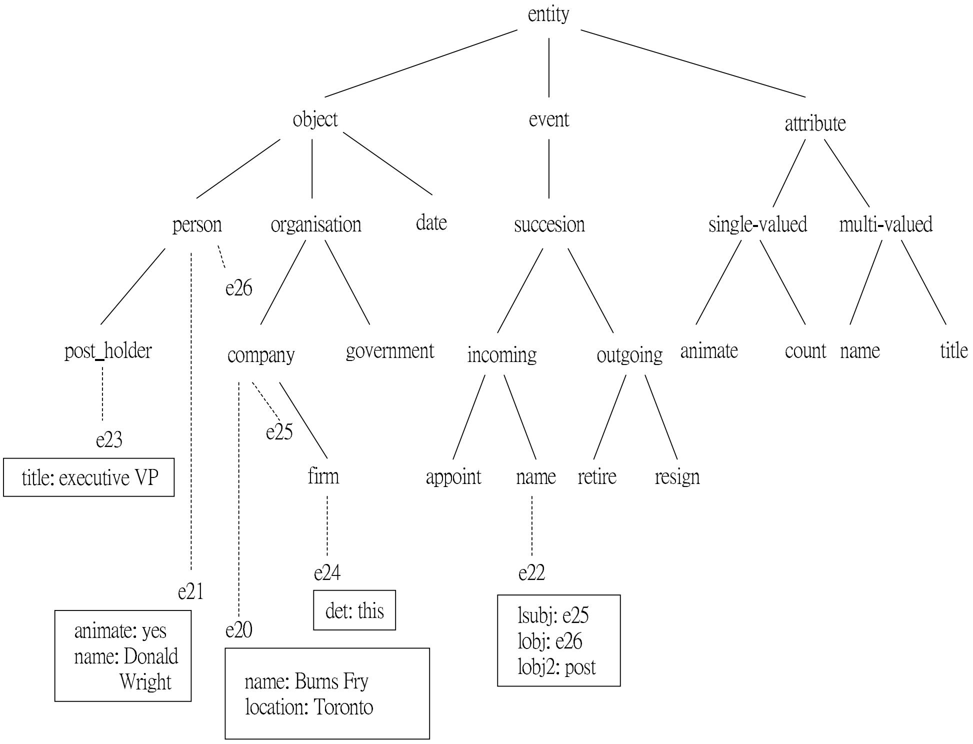


Figure 5 *A Fragment of the LaSIE Discourse Model*

Discourse processing proceeds in four substages for each new sentence representation passed on from the parser. First, the semantic representation producedby the parser is processed by adding its instances, together with their attributes, to the discourse model which has been constructed so far for the text. Instances which have their semantic class specified in the input (via unary predicates) are added directly to the discourse model, beneath their class in the ontological hierarchy (e.g. firm(e24)). Attributes -- binary predicates in which the first argument is always an instance identifier -- are added to the attribute-value structure associated with instance identifiers occurring within them, provided the class of the instance is known.

In the second stage, presuppositions are expanded, leading to further information being added to or removed from the model. In the current example, this has two effects. First, it permits missing semantic class information for instances to be derived from type restrictions on attribute arguments. For instance,an **attr_of** attribute associated with the node in the ontology corresponding to the **title** attribute, records that this attribute holds only of entities of type **post_holder**. Thus, given the input fact **title(e23,executive VP)** but no input fact specifying the class of **e23**, it becomes possible to attach the instance **e23** beneath the correct class in the ontology. Second, the semantic types of verbal roles are used to hypothesise entities which fulfil those roles, if they are not present, or have not been discovered, in the input. In this case the fact that `Donald Wright' is the logical object of the `was named' event has not been determined by the parser, as the intervening phrase `46 years old' was not properly parsed, hence preventing theparser from identifying `Donald Wright' as the surface subject/logical object of the passive verb phrase. Thus, a person **e26** is added to the model to play this role. In a similar fashion **e25**, an organisation, is added to the model to play the role of the logical subject of the naming event.

The third stage involves comparing all new instances (those introduced by thissentence) with previously existing instances to determine whether any pair can be merged into a single instance, representing a coreference in the text. The algorithm takes into account considerations such as the instances' textual proximity and the consistency of their semantic classes and attributes. For the current example the coreference algorithm leads to the merging of **e26** and **e21** -- that is, `Donald Wright' is recognised as the logical object of the naming event -- and **e25** is merged with **e24** -- that is, `this brokerage firm' is identified as the logical subjectof the naming event. Subsequently these merged entities are merged with **e20** -- that the brokerage firm doing the naming is identified as `Burns Fry'. The reader is referred to (5) for further details, and an evaluation, of the coreference algorithm.

The final stage of discourse processing is consequence expansion. This stage is intended to allow any inferences to be drawn which can now be made given the addition to the discourse model of the information in the current sentence. Its primary use in

LaSIE is to allow inference rules associated with template objects and slots to infer values for these objects and slots from information now present in the discourse model.

After all sentences in a text have been processed, the template will have beenfilled to the best of the system's abilities. The template is then written out in whatever form is required.

3.3 Trends

IE is not an isolated activity and is being influenced by and is in turn influencing other activities in natural language processing and computational linguistics. In this section we look briefly at three trends that can be seen in the recent development of IE: the movement towards shallower processing (or towards what might be called an `appropriate' level of processing for the task), the movement away from handcrafted rule sets towards automatically acquired rule sets, and the movement towards coupling together relatively independent modules. Of course these trends are not entirely independent. They are all part of a general move towards a more empirically oriented approach to NLP that has emerged for a host of reasons, including the availability of large scale electronic corpora, frustration with theoretical developments that seemed to be losing touch with the reality of the data, and the drive towards applications.

3.3.1 Shallow vs Deep Processing

Given the pragmatic constraints imposed by the IE task -- the relatively limited understanding required -- many developers of IE systems have, in recent years, opted for engineering solutions that de-emphasize the substantial body of theoreticalwork both in computational syntax and semantics and in knowledge representation and reasoning. This de-emphasis is perhaps most dramatically illustrated by SRI who abandoned, quite consciously, the theoretically motivated TACITUS system after MUC-3 (1991) in favour of the pragmatically motivated FASTUS system which they have used for MUC-4 (1992) through MUC-6 (1995). TACITUS (5) attempted a full syntactic analysis, using a large scale grammar of English, performed semantic interpretation to produce first-order predicate calculus representations, and then used abductive reasoning to interpret the semantic representations of individual sentences in the context of a schema pertaining to the scenario of interest. FASTUS (5), by contrast, uses a cascade of finite-state transducers that successively tokenise, recognise names, recognise phrases, recognise template patterns, and then combine or merge partially filled templates to generate the final template. SRI have been keen to stress that this change in direction has not happened because they concluded that the TACITUS approach was faulty, but because they believed it was inappropriate for the task. TACITUS did text understanding, FASTUS information extraction, the latter, on their view, a much simpler task that does not require the theoretical and computational sophistication of TACITUS. The chief gain from the switch has been speed (from 36 hours to 12 minutes for 100 texts between MUC-3 and MUC-4) and to some extent ease ofporting to new domains. Though performance results, in terms of combined precision and recall, are not strictly comparable between MUCs, it is worth noting that FASTUS scores surpassed TACITUS scores by about 16% between MUC-3 and Muc-4, mostly due to increased recall.

SRI have not been alone in moving away from a more powerful, linguistically motivated approach towards a more restricted, task-specific, engineering-driven approach. Recent IE systems developed by General Electric, Mitre Corporation, New York University and SRA have all come to be considered exemplars of a `shallow' processing approach to IE which promises, if not better recall and precision, at least faster, more portable systems.

This movement away from the more theoretically motivated work of the 1980's has engendered considerable debate (and rhetoric) about `shallow' versus `deep' approaches to information extraction. This debate is ongoing and the underlying distinction, while reflecting important insights, needs to be analysed, as it can lead to distortion and over-simplification. In particular, it is important to distinguish at least two ways in which processing in an IE system can be shallower or deeper. The processing in an IE system can be divided coarsely into two parts: the syntactic portion that works on single sentences of the input and the discourse-level portion that integrates information from the syntactic analyses of multiple sentences. The former typically includes tokenisation, part-of-speech tagging, phrasal pattern matching or parsing and produces a regularised form which may be anything from a partially filled template to a full logical form.The latter takes whatever regularised form has been produced by the former and, perhaps using more general knowledge of domain, attempts to integrate information from the individual sentence representations into a larger scale structure which ultimately either is, or serves to provide, the information for the final template.

Thus, processing in an IE system can be shallower or deeper depending on the shallowness or depth of processing in each of these two processing stages. First, the syntactic analysis the system performs can be more or less thorough. At one extreme there are systems which employ formally weak mechanisms (finite-state pattern matchers) to apply domain-specific lexically-triggered patterns; at the other extreme there are systems which employ formally stronger mechanisms (complete parsers for context-free or even more expressive formalisms) to apply general grammars of natural language. Examples of the former include the SRI FASTUS system, Mitre's Alembic} system (5), and the SRA (5) and NYU (5) MUC-6 systems; examples of the latter include the TACITUS system mentioned above, the Proteus system (5), and the PIE system (5). Systems like LaSIE and the BBN PLUM system (5) which use a domain independent grammar, but only attempt fragmentary parsing, fall somewhere in the middle.

Second, the discourse or multi-sentence level processing can be more or less general. Thus, the semantic representation derived from the syntactic analysis can be expressed in a more or less general formalism and manipulated by more or lessgeneral algorithms which attempt to integrate it into a more or less general model of the text and domain. There may or may not be any attempt to use declaratively represented world and domain knowledge to help in resolving ambiguities of attachment, word sense, quantifier scope, and coreference, or to support inference-driven template filling. At one extreme there are information extraction systems which produce semantic representations that are fragments of the target template for just those sentences that yield template relevant information and then merge these using *ad ho*c heuristics to produce the final template (e.g. FASTUS and the SRA MUC-6 system); at the other extreme there are systems that use abductive theorem provers and axiomatisations of the domain to compute the least cost explanation of the first order logic expressions derived from every sentence in the input and then generate the template from the resulting underlying logical model (e.g. TACITUS). In between lie systems that translate their input into some sort of template-independent predicate-argument notation and use some amount of declaratively represented information about the domain to assist in doingcoreference and inference driven template filling. LaSIE falls into this camp as do the NYU MUC-6 system and the MITRE Alembic system.

3.3.2 Hand-crafted Rules vs Automated Rule Acquisition

Early successful systems like JASPER (see section 2.1 above), depended on very Complex hand-crafted templates, made up by analysts. However, the IE movement has grown by exploiting, and joining, the recent trend towards a more empirical and text-based computational linguistics, that is to say by putting less emphasis on linguistic theory and trying to derive structures and various levels of linguistic generalisation from the large volumes of text data that machines can now manipulate.

A conspicuous success has been part-of-speech taggers, systems that assign one and only one part-of-speech symbol to a word in a running text and do so onthe basis (usually) of statistical generalisations across very large bodies of text. Recent research has shown that a number of quite independent modules of analysisof this kind can be built up independently from data, usually very large electronic texts, rather than coming from either intuition or some dependence on other parts of a linguistic theory. These independent modules, each with reasonably high levels of performance in blind tests, include part-of-speech tagging, aligning texts sentence-by-sentence in different languages, syntax analysis, and attaching word sense tags to words in texts to disambiguate them in context.

The empirical movement, basing, as it does, linguistic claims on text data, hasanother stream: the use in language processing of large language dictionaries (of single languages and bilingual forms) that became available about ten years ago in electronic forms from publishers' tapes. These are not textual data in quite the sense above, since they are large sets of intuitions about meaning set out by teamsof lexicographers or dictionary makers. Sometimes they are actually wrong, but they have nevertheless proved a useful resource for language processing by computer, and lexicons derived from them have played a role in actual working MT and IE systems (5).

What such lexicons lack is a dynamic view of a language; they are inevitably fossilised intuitions. To use a well known example: dictionaries of English normally tell you that the first, or main, sense of ``television'' is as a technology or a TV set, although it is mainly used now to mean the medium itself. Modern texts are thus out of step with dictionaries -- even modern ones. It is this kind of evidence that shows that, for tasks like IE, lexicons must be adapted or ``tuned'' tothe texts being analysed which has led to a new, more creative wave, in IE research: the need not just to use large textual and lexical resources, but to adapt them as automatically as possible, to enable them to adapt to new domains and corpora, which will mean dealing with obsolescence and with the specialised vocabulary of a domain not encountered before.

3.3.3 Modularisation

As noted above there has been a movement away from theory prescribed modules whose processing is controlled by sets of handcrafted rules towards data-dependent modules whose processing is controlled by rules or parameters acquiring from automatically analysing large text corpora. These modules include part-of-speech tagging, text-alignment in different languages, syntax analysis, word sense disambiguation and so on. Aside from the fact that their rules or parameters are acquired automatically, the other striking thing about these modules is their independence: that these tasks can be done relatively independently is very surprising to those who believed them all contextually dependent sub-tasks within alarger theory. These modules have been combined in various ways to perform taskslike IE as well as more traditional ones like machine translation (MT). The modules can each be evaluated separately -- against their specifications. Recently there has been a move to support this kind of modularisation explicitly through the development of text processing architectures like the TIPSTER architecture (5) and implementations of it like the General Architecture for Text Engineering (GATE) (5;5). These architectures support rapid addition and interchange of modules and represent a commitment to a modular approach to language engineering.

While language engineering modules can be developed and evaluated independently it is important to keep in mind that they do not in the end do tasksthat real people actually do, unlike MT and IE systems. One can call the former `intermediate' tasks and the latter real or final tasks -- and it is really only the latter that can be firmly evaluated against human needs -- by people who know what a translation, say, is and what it is for. The intermediate tasks are evaluated internally to improve performance but are only, in the end, stages on the way to some larger goal. Moreover, it is not possible to have quite the same level of confidence in them since what is, or is not, a correct syntactic structure for a sentence is clearly more dependent on one's commitments to a linguistic theory of some sort, and such matters are in constant dispute. What constitutes proper extraction of people's names from texts, or a translation of it, can be assessed by many people with no such subjective commitments.

4. Application Areas of Information Extraction

In section 2 we reviewed work in IE from an historical perspective, describing efforts in the area in a chronological fashion. It is also of interest, however, to view IE from the perspective of the application areas in which IE systems have been or are being deployed. This perspective should help to dispel the view, whichthe MUC evaluations may have unintentionally engendered, that IE is only of interest for military intelligence or financial applications, and to stimulate thinking about the range of potential applications for this growth technology.

The following list is bound to be partial; but it is indicative of the range of areas in which IE technology is already in play.

Finance The MUC-5 joint ventures scenario lead at least thirteen sites to develop IE systems for extracting details of joint ventures from newswire stories (5). The MUC-6 management succession event scenario is also of potential interest to those working in finance (5). The COBALT and FACILE projects (5;5) which use IE techniques to help categorise newswire stories of relevance to stock traders also operate in this area. A number of companies have expressed interest to the authors in competitor intelligence systems that will enable them to track ventures in which their competitors are engaged, as reported in newswires.

Military intelligence The U.S.\ Air Force supported early research on the extraction of satellite events (5). MUC-1 and MUC-2 focussed on hostile actions of enemy units against U.S. naval forces. MUC-3 and MUC-4 concentrated on gathering information about terrorist attacks from Latin American newsfeeds (5;5).

Medicine Sager's early work (5) illustrated the possibility of gathering information from patient discharge summaries and radiology reports. Work by Lehnert also applied IE in a medical domain (5). We have discussed applications of IE with local medical informatics experts and they confirm the need for applications to help in the classification of patient records and discharge summaries to assist in public health research and in medical treatment auditing.

Law The NAVILEX project aims to use IE techniques to support intelligent retrieval from legal texts (5). It follows on from the NOMOS project which also applied `shallow' NLP techniques to extract information from legal texts to assist in retrieval (5).

Police The POETIC project developed an IE system for extracting information about road traffic incidents from police `command and control' incident logs (5). The AVENTINUS project is working to build tools to assist police in criminal investigations relating to drug trafficking (5;5).

Technology/product tracking One of the two MUC-5 extraction scenarios was microelectronics products announcements -- extracting details about new microelectronic technology from the trade press (5). Again, industrialists have expressed an interest to us in tracking commodity price changes and factors affecting these changes in the relevant newsfeeds.

Academic research Academic journals and publications are increasingly becoming available on-line and offer a prime, if challenging, source of material for IE technology. The EMPathIE project in which we are currently involved is exploring the possibility of building an Enzyme and Metabolic Pathways database using IE techniques to fill in templates about enzymes and enzyme activities from electronic versions of relevant biomolecular journals (5). Cowie's work on wild flower guides (5) and Zarri's work on historical texts (5) are early examples of this sort of work.

Employment The TREE project aims to build a database of employment opportunities from electronic job advertisements (5;5).

Fault Diagnosis The SINTESI project extracts information from reports of car faults (5); the TACITUS system was also employed in analysing engine failure reports (5;5).

Software system requirements specification NLP techniques have been used to assist in the process of deriving formal software specifications from less formal, natural language specifications. We are currently involved in research to see if this problem can be cast in the form of an IE problem, where the formal specification is viewed as a template which needs to be filled from a natural language specification, supplemented with a dialogue with the user.

Together these applications demonstrate the broad range of projects already undertaken or in progress which utilise IE technology. Clearly they represent but a tiny fraction of potential applications -- which supports our claim to the importanceof IE as a growth text processing technology.

5. Concluding Remarks

5.1 Challenges for the Future

We hope the foregoing discussion has illuminated the objectives of IE, the as yet brief history of this area of research, the sorts of approaches that are being used, and the areas of application which have been and are being considered. In concluding we focus on a number of central challenges facing IE in the future.

5.1.1 Higher Precision and Recall

Combined precision and recall scores for IR systems have rested in the mid-50% range for many years, and it is in this range that current IE systems also find themselves. While users of IR systems have adapted themselves to these performance levels, it is not clear that for IE applications such levels are acceptable. Clearly what is tolerable will vary from application to application. But where IE applications involve building databases over extended periods of time which subsequently form the input to further analysis, noise in the data will seriously compromise its utility. Cowie and Lehnert (5) suggest that 90% precision will be necessary for IE systems to satisfy information analysts. Currenthigh precision scores in the MUC scenario extraction tasks are around 70%.

Improvements in both precision and recall are high priority challenges for IE systems. There are no `magic bullets' on the horizon, but there is every reason to believe that significant progress can be made as research continues in NLP and asmore lexical and grammatical resources become available.

5.1.2 User-defined IE

Currently IE systems are tailored for new applications through a two stage processwhich involves first defining a template for the application -- identifying the entities, attributes and relations to be captured -- and second modifying the lexical,grammatical and conceptual rule-bases that the IE system uses in carrying out its text processing. Both of these stages typically require the involvement of experts. The first requires a logical analysis of the information to be captured and the articulation of this analysis in a particular formalism. Given that the second stage of the customisation is highly dependent on this first stage and will require considerable effort, it is important that this stage be carried out correctly and giventhe current development of the technology this is only probable if the person defining the template has a good grasp of the nature and limits of IE systems.

The second stage of customisation -- modifying the lexical, grammatical and conceptual rule-bases that the IE system uses in carrying out its text processing -- clearly requires expert knowledge. If these rule-bases are handcrafted, then those with the knowledge to do the handcrafting -- typically computational linguists or NLP experts -- must perform the customisation for each new domain. If the rule-bases are not handcrafted, but acquired from corpora, then the corpora must be carefully selected, perhaps annotated, and the rule acquisition process monitored carefully.

Thus porting IE systems to new domains is a serious bottleneck for state-of-the-art systems. As a consequence, the development of IE technology that permits users to define the extraction task and then adapts to the new scenario is a major challenge: only with the development of such user-centred, adaptive systems is IE technology likely to become of utility to information gathers other than those who can afford to dedicate months of expensive customisation effort to the task.

Some progress has been made in this direction. The final MUC-6 scenario task was only given to participants one month before the evaluation in an effort toreward highly portable systems. SRA have begun developing tools to help users define templates through examples (5). Morgan *et al.* (5) have also experimented with various techniques to allow users to customise the Lolita system for new IE tasks.

5.1.3 Integration with other Technologies

IE need not be considered a standalone technology which is of use only for applications in which a structured database is to be created from a text corpus. There are a number of other technologies with which it might be combined to yield powerful new information gathering capabilities.

Information Retrieval The TIPSTER programme from the very start conceived of IR and IE asnaturally forming two stages of a coupled information gathering effort, referring to them as detection and extraction respectively. The assumption was that an initial user query would be given to an IR system which from a potentially massive document collection would detect the relevant documents to be passed on to an IE system for the more detailed and computationally intensive analysis that such systems carry out.

While this coupling was initially conceived of in the context of the massive electronic document collections being assembled by governments and other large organisations, the arrival of the WWW has made available a document collection whose size threatens to dwarf anything the TIPSTER convenors conceived of as little as five years ago.

Despite the natural complementarity of IR and IE we are not aware of much practical work which has gone on in this direction as yet. We have done some preliminary experimental work in using Web search engines to create document collections which are then processed by the LaSIE system, and are encouraged by the results (5;5). However much more work needs to be done in this area, and no doubt will be.

Aside from this obvious way of combining IR and IE systems, there are other-possible ways in which the two technologies may be of mutual benefit. Specifically, for applications where the computational intensiveness of IE systems isnot a drawback, an IE system could be used in conjunction with the indexing component of an IR system in one of a number of ways. Most obviously, the proper name recognition and classification abilities of an IE system could be harnessed to provide useful (possibly) multi-word, preclassified index terms that would enable searches for, e.g., `Ford' the company, and exclude all references to persons and places named `Ford'. But more sophisticated indexing could be developed based on the identification of entities and relations, such as IE systems carry out. For example, remaining with the management succession scenario, one could index documents according to succession events and roles in them so that one could search for all reports mentioning persons who had resigned from CEO positions in Canadian companies in the last year. Work on using IE templates for indexing legal documents is implemented in the Navilex system (5); work on usingIE techniques to supplement traditional IR approaches to categorising and filtering news stories is being carried out in the related COBALT and FACILE projects, as mentioned above in section 2.3. Clearly there are many further potential applications of this nature.

Natural Language Generation Our example in Figure 1 showed the NL summarythe LaSIE system generated from the template it had extracted. This summary was generated using very crude generation techniques. Given that much more sophisticated NL generation (NLG) capabilities now exist (5), the coupling of IE and NLG should permit more fluid, easy to read summaries to be generated from extracted templates.

Machine Translation The translation of documents may be carried out for many reasons, but if the purpose of the translation is to enable subsequent extraction of information from the text that was previously inaccessible to the information seeker because of the language barrier, then given the difficulty of translation it isworth considering ways in which the information sought could be first extracted and then translated. That is, rather than performing translation followed by extraction, it may be preferable to perform extraction in the source language followed by translation into the destination language. Such a coupling of IE and MT technologies is particularly attractive because a template, being regularised provides a much easier information source to translate than a full text.

Some work along these lines has already been carried out (5;5) but we expectmuch more work to be carried out in this area in the near future. Again, given the sudden availability of multilingual on-line text afforded by the Web, information gatherers will want ways of accessing this information that avoid the overheads of large scale translation.

Data Mining IE systems produce structured data repositories which can be turned into conventional databases to be accessed with conventional database access tools such as SQL query processors. However, these databases may also be processed bydata mining (DM) or knowledge discovery in database (KDD) tools which seek novel patterns in the data (5). The significance of coupling IE with DM or KDD techniques is that this will permit hitherto unmined text resources to become the subject of extensive exploration. As one example, consider the possibilities of extracting information about commodity price changes from financial news reports, building a database of these fluctuations over some historical period and then usingKDD techniques to discover correlations that might give insights into the causes ofthese changes. Once again, coupling IE with another technology promises powerful new techniques for gathering information from texts.

5.2 IE or not IE?

An important insight, even after accepting our argument that IE is a new, emergent technology, is that what may seem to be wholly separate information technologies are really not so: MT and IE, for example, are just two ways of producing information to meet people's needs and can be combined in differing ways: for example, one could translate a document and then perform IE against the result or vice-versa, which would mean just translating the contents of the resulting templates. Which of these one chose to do might depend on the relativestrengths of the translation systems available: a simpler one might only be adequateto translate the contents of templates, and so on. This last observation emphasizesthat the product of an IE system -- the filled templates -- can be seen either as acompressed, or summarised, text itself, or as a form of data base (with the fillers of the template slots corresponding to conventional database fields). One can then imagine new, learning, techniques like data mining being done as a subsequent stage on the results of IE itself.

If we think along these lines we see that the first distinction of this paper, between traditional IR and the newer IE, is not totally clear everywhere but can itself become a question of degree. Suppose parsing systems that produce syntactic and logical representations were so good, as some now believe, that they could process huge corpora in an acceptably short time. One can then think of the traditional task of computer question answering in two quite different ways. The old way was to translate a question into a formalised language like SQL and use it to retrieve information from a database -- as in `Tell me all the IBM executivesover 40 earning under £ 50K a year'. But with a full parser of large corpora one could now imagine transforming the query to form an IE template and searching the whole text (not a data base) for all examples of such employees -- both methods should produce exactly the same result starting from different information sources -- a text versus a formalised database.

What we have called an IE template can now be seen as a kind of frozen query that one can reuse many times on a corpus and is therefore only important when one wants stereotypical, repetitive, information back rather than the answer toone-off questions.

Tell me the height of Everest, as a question addressed to a formalised text corpus is then neither IR nor IE but a perfectly reasonable single request for an answer. `Tell me about fungi', addressed to a text corpus with an IR system, will produce a set of relevant documents but no particular answer. `Tell me what films my favourite movie critic likes', addressed to the right text corpus, is undoubtedly IE, and will produce an answer also. The needs and the resources available determine the techniques that are relevant, and those in turn determine what it is to answer a question as opposed to providing information in a broader sense.

Acknowledgments

Thanks to Beth Sundheim of the US Naval Command, Control, and Ocean Surveillance Center RDT\&E Division (NRaD) for detailed comments during the preparation of this paper.

The University of Sheffield's LaSIE system and the GATE architecture described here were supported by the UK EPSRC (grant number GR/K25267).

References

WITTEN, I.H., MOFFAT, A., & BELL, T.C. *Managing Gigabytes*. New York: Van Nostrand Reinhold, 1994.

JACOBS, P.S., & RAU, L.F. SCISOR: Extracting information from on0line news. *Communications of the ACM*, 33(11), 1990, 88-97.

CIRAVEGNA, F., CAMPIA, P., & COLOGNESE, A. Knowledge Extraction from Texts by SINTESI. In: *Proceeding of the Fifteenth International Conference on Computational Linguistics* (COLING-92). 1992, 1244-1248.

COWIE, J., & LEHNERT, W. Information Extraction. *Communications of the ACM*, 39(1), 1996, 80-91.

SAGER, N. *Natural Language Information Processing*. Reading, Massachusetts: Addison Wesley, 1981.

COLLER, R. A*utomatic Template Creation for Information Extraction*. Technical Report CS-96-07. Department of Computer Science, University of Sheffield, UK, September 1996.

SCHANK, R.C., & COLBY, M.C. *Computer Models of Thought and Language*. San Francisco: W.H. Freeman, 1973.

SCHANK, R.C. *Conceptual Information Processing*. Amsterdam: North-Holland, 1975.

SCHANK, R.C., & ABELSON, R.P. *Scripts, Plans, Goals, and Understanding*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1977.

DEJONG. G. An Overview of the FRUMP System. In: LEHNERT, W., & RINGLE, M.h. (eds), *Strategies for Natural Language Processing*. *Lawrence Erlbaum*, 1982, 149-176.

LYTINEN, S.L., & GERSHMAN, A. ATRANS: Automatic processing of money transfer messages. *In: Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*. 1986, 1089-1093.

ANDERSEN, P.M., HAYES, P.J., HUETTNER, A.K., NIRENBURG, I.B., SCHMANDT, L.M. & WEINSTEIN, S.P. Automatic Extraction of Facts from Press Releases to Generate News Stories. *In: Processing of the Third Conference on Applied Natural Language Processing*. 1992, 170-177.

COWIE, J.R. Automatic Analysis of Descriptive Texts. *In Proceedings of the ACL Conference on Applied Natural Processing*. 1983, 117-123.

ZARRI, G.P. Automatic Representation of the Semantic Relationships Corresponding to a French Surface Expression. *In: Proceedings of the ACL Conference on Applied Natural Language Processing*. 1983, 143-147.

Proceedings of the Third Message Understanding Conference (MUC-3). Morgan Kaufmann, for Defense Advanced Research Projects Agency, 1991.

Proceedings of the Fourth Message Understanding Conference (MUC-4). Morgan Kaufmann, for Defense Advanced Research Projects Agency, 1992.

Proceedings of the Fifth Message Understanding Conference (MUC-5). Morgan Kaufmann, for Defense Advanced Research Projects Agency, 1993.

Proceedings of the Sixth Message Understanding Conference (MUC-6). Morgan Kaufmann, for Defense Advanced Research Projects Agency, 1995.

SUNDHEIM, B. Tipster/MUC-5 Information Extraction System Evaluation. *In: Proceedings of the Fifth Message Understanding Conference (MUC-5).* Morgan Kaufmann, 1993, 27-44.

VAN RIJSBERGEN, C.J. *Information Retrieva*l. London: Butterworths, 1979.

VILAIN, M., BURGER, J., ABERDEEN, J., CONNOLLY, D., & HIRSCHMAN, L. A Model Theoretic Coreference Scoring Scheme. *In: Proceedings of the Sixth Message Understanding Conference (MUC-6).* Morgan Kaufmann, 1995, 45-52.

CHINCHOR, N. The Statistical Significance of the MUC-5 Results. *In: Procedings of the Fifth Message Understanding Conference (MUC-5).* Morgan Kaufmann, 1993, 79-83.

MILLER, G.A. WordNet: An on-line Lexical Database. *International Journal of Lexicography*, 3(4), 1990, 235-3112.

Evans, R., GAIZAUSKAS, R., CAHILL, L., WALKER, J., RICHARDSON, J., & DIXON, A. POETIC: A System for Gathering and Disseminating Traffic Informtaion. *Journal of Natural Language Engineering*, 1(4), 1995, 363-387.

ELLMAN, J., SOMERS, H., NIVRE, J., & MULTARI, A. Foreign Language Information Extraction: An Application in the Employment Domain. *In: Natural Language Processing: Extracting Information for Business Needs*. Unicom Semantics Ltd., London, March 1997. 77-89.

TREE: *Trans European Employmen*t. http://www2.echo.lu/language/en/lel/tree/tree.html. Site visited 29/05/97.

ROCCA, G., SPAMPINATO, L., ZARRI, G.P., BLACK, w., & CELNIK, P. COBALT: Construction, Augmentation and Use of Knowledge bases from Natural Language Documents. *In: Proceedings of the Artificial Intelligence Conference*. May 1994.

BLACK, W.J. FACILE: Fine-Grained Multilingual Text Categorisation and Information Extraction, *In: Natural Language Processing: Extracting Information for Business Needs*. Unicom Seminars Ltd., London, March 1997, 119-131.

FACILE: *Fast and Accurate Categorisation of Information by Language Engineering*.

http://www2.echo.lu/langeng/en/lel/facile/facile.html. Site visited 29/05/97.

THURMAIR, G. Information Extraction for Intelligence Systems. I*n: Natural Language Processing: Extracting Information for Business Needs*. Unicom Seminars Ltd., London, March 1997, 135-149.

AVENTINUS: *Advanced Information Systems for Multinational Drug Enforcement*. http://www2.echo.lu/langeng/en/lel/aventinus/aventinus/aventinus.html. Site visited 29/05/97/

ECRAN: *Extraction of Content: Research at Near-Market*. http://www2.echo.lu/langeng/en/lel/ecran.ecran.html. Site visited 29/05/97.

HOBBS, J.R. The Generic Information Extraction System. *In: Proceedings of the Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann, 1993, 87-91.

GAIZAUSKAS, R.G., CUNNUNGHAM, H., WILKS, Y., RODGERS, P., & HUMPHREYS, K. GATE- an Environment to Support Research and Development in Natural Language Engineering. *In: Proceedings of the 8th IEEE International Conference on Tools with Artificial Intelligence (ICTAI-96)*. October 1996, 58-66.

CUNNINHAM, H., HUMPHREYS, K., GAIZAUSKAS, R., & WILKS, Y. Software Infrastructure for Natural Language Processing. *In: Procedings of the Fifth Conference on Applied Natural Language Processing (ANLP-97)*. Available at http://xxx.lan/.gov/ps/9702005. March 1997, 237-244.

GRISHMAN, R. TIPSTER A*rchitecture Design Document Version 2.2*. Technical Report. Defense Advanced Research Projects Agency. Available at http://www.tipster.org/. 1996.

HUMPHREYS, K., GAIZAUSKAS, R., CUNNUNGHAM, H., & AZZAM, s. VIE *Technical Specifications*. Department of Computer Science, University of Sheffield, 1996.

BRILL, E. A Simple rule-based part-of-speech tagger. *In: Proceeding of the Third Conference on Applied Natural Language Processing*. 1992, 152-155.

MARCUS, M.P., SANTORINI, B., & MARCINKIEWICZ, M.A. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2), 1993, 313-330.

WAKAO, T., GAIZAUSKAS, R., & WILKS, Y. Evaluation of an Algorithm for the Recognition and Classification of Proper Names. *In: Proceedings ot the 16th International Conference on Computational Linguistics (COLING96)*. 1996, 418-423.

GAZDAR, G., & MELLISH, C. *Natural Language Processing in Prolog*. Wokingham: Addison Wesley, 1989.

GAIZAUSKAS, R. *Investigations into the Grammar Underlying the Penn Treebark II* . Technical Report CS-95-25. Department of Computer Science, University of Sheffiled,1995.

GAIZAUSKAS, R., & X I : *A Knowledge Representation Language Based on Cross-Classification and Inheritance*. Technical Report CS-95-24. Department of Computer Science, University of Sheffield. 1995.

GAIZAUSKAS, R., & HUMPHREYS, K. Quantative Evaluation of Coreference Algorithms in an Information Extraction System. In: BOTLEY, s., & MCENERY, T. (eds), *Discourse Anaphora and Anaphor Resolution*. University College London Press, 1997, (in press).

HOBBS, J.R. Description of the TACITUS System as Used for MUC-3. *In: Proceedings of the Third Message Understanding Conference MUC-3*. Morgan Kaufmann, 1991,200-206.

HOBBS, J.R., APPELT, D., TYSON, M., BEAR, J., & ISRAEL, D. Description of the FASTUS System as Used for MUC-4. *In: Proceedings of the Fourth Message Understanding Conference MUC-4*. Morgan Kaufmann, 1992, 268-275.

APPELT, D.E., HOBBS, J.R., BEAR, J., ISRAEL, D., KAMEYAMA, M., & TYSON, M. Description of the JV-FASTUS System as Used for MUC-5. *In: Proceedings of the Fourth Message Understanding Conference MUC*-5. Morgan Kaufmann, 1993, 221-235.

APPELT, D., HOBBS, J., BEAR, J., ISREAL, D., KAMEYAMA, M., KEHLER, A., MARTIN, D., MYERS, K., & TYSON, M. SRI International FASTUS system: MUC-6 Test Results and Analysis. *In; Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, 1995, 237-248.

ABERDEEN, J., BYRGER, J., DAY, D., HIRSCHMAN, L., ROBINSON, P., & VILAIN, M. MITRE: Description of the Alembic System Used for MUC-6. *In: Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, 1995, 141-156.

KRUPKA, G.R., Description of the SRA System as used for MUC-6. *In: Proceedings of the Fourth Message Understanding Conference (MUC-6*). Morgan Kaufmann, 1995, 221-236.

GRISHMAN, R. TIPSTER *Architecture Design Document Version 1.52 (Tinman Architecture)*. Technical Report. Departmnet of Computer Science, New York University. Available at http://www.cs.nyu.edu/tipster. 1995.

GRISHMAN, R., & STERLING, J. Description of the Proteus System as Used for MUC-5. *In: Proceedings of the Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann, 1993, 181-194.

LIN, D. Description of the PIE System as used for MUC-6. *In: Proceedings of the Sixth Message Understanding Conference (MUC-6)*. San Francisco: Morgan Kaufmann. 1995, 113-126.

WEISCHEDEL, R. Description of the PLUM System as used for MUC-6. *In: Proceedings of the Sixth Message Understanding Conference (MUC-6)*. San Francisco: Morgan Kaufmann. 1995, 55-70.

WILKS. Y., GUTHRIE, L., & SLATOR, B. *Electric Words*. Cambridge, MA: MIT Press. 1996.

GRISHMAN, R., & SUNDHEIM. B. Message Understanding Conference - 6: A Brief History. *In: Proceedings of the ACL Conference on Applied Natural Language Processing*. 1983.

LEHNERT, W., SODERLAND, S., ARONOW, D., FENG, F., & SHMUELI, A Inductive Text Classification for Medical Applications. *Journal for Experimental and Theoretical Artificial Intelligence*, 7(1), 1994, 49-80.

PIETROSANTI, E., & GRAZIADIO, B. Artificial Intelligence and Legal Text Management: Tools and Techniques for Intelligent Document Processing and Retrieval. *In: Natural Language Processing: Extracting Information for Business Needs*. Unicom Seminars Ltd., London, March 1997, 277-291.

GIANETTI, A., DASSOVICH, P., MARCHIGNOLI, G., PIETROSANTI, E., AZZAM, s., CELNIK, P., BILON, J., FORTIER, V., & PIRES, F. NOMOS: Knowedge Acquisition for Normative Engineering Reasoning Systems. In: STEELS, L., & LEPAPE, B. (eds), *Enhancing the Knowledge Engineering Process: Contributes from ESPRIT. Elsevier Science Publications*, 1992.

EMPathIE: *Enzyme and Metabolic Path Information Extraction*.
http://www.dcs.shef.ac.uk/research/groups/nlp/funded/empathie.html.
Site visited 29/05.97.

HOBBS, J.R., The TACITUS Project. *Computational Linguistics*, 12(3), 1986, 220-222.

HOBBS, J.R., STICKEL, M.E., APPELT, D.E., & MARTIN, P. Interpretation as Abduction. *Artificial Intelligence*, 63, 1993, 69-142.

MORGAN, R.G. *An architecture for user defined information extraction*. Technical Report 8/96. Department of Computer Science, University of Durham, 1996.

ROBERTSON, A.M., & GAIZAUSKAS, R. On the Marriage of Information Retrieval and Information Extraction. In: FURNER, J., & HARPER, D.J., (eds), *Proceedings of the 19th Annual BCS-IRSG Colloquium on IR Research*. 1997, 60-67.

GAIZAUSKAS, R., & ROBERTSON, A.M. Coupling Information Retrieval and Information Extraction: A New Text Technology for Gathering Information from the Web. *In: Proceedings of the 5th Computed-Assisted Information Searching on Internet Conference (RIAO'97)*. 1997, 356-370.

UZKOREIT, H. Language Generation. *In: Survey of the State of the Art in Human Language Technologies*. http://www.cse.ogi.edu/CLSU/HLTsurvey/HLTsurvey.html: Centre for Spoken Language Understanding, Oregon Graduate Institute. 1997.

KAMEYAMA, M. Information Extraction across Linguistic Barriers. *In: AAAI Spring Sympo-*

sium on Cross-Language Text and Speech Retrieval. March 1997.

AZZAM. S., HUMPHREYS, K., GAIZAUSKAS, R., CUNNUNGHAM, H., & Wilks, Y. Using a Language Independent Domain Model for Multilingual Information Extraction. In: SPRYRODOPOULOS, C. (ed), *Proceedings of the IJCAI-97 Workshop on Multilinguality in the Software Industry*: the AI Contribution (MULSAIC-97). 1997, (in press).

FAYYAD, U., PIATETSKY-SHAPIRO, G., & SMYTH, P. From Data Mining to Knowledge Discovery in Databases. *AI Magazine*. 1996, 37-54.

Computational Linguistics and Chinese Language Processing
vol. 3, no. 2, August 1998, pp. 61-78
©Computational Linguistics Society of R.O.C.

An Assessment of Character-based Chinese News Filtering Using Latent Semantic Indexing

Shih-Hung Wu*, Pey-Ching Yang*, Von-Wun Soo*

Abstract

We assess the Latent Semantic Indexing (LSI) approach to Chinese information filtering. In particular, the approach is for Chinese news filtering agents that use a character-based and hierarchical filtering scheme. The traditional vector space model is employed as an information filtering model, and each document is converted into a vector of weights of terms. Instead of using words as terms in the IR nominating tradition, terms refer to Chinese characters. LSI captures the semantic relationship between documents and Chinese characters. We use the Sin-gular-value Decomposition (SVD) technique to compress the term space into a lower dimension which achieves latent association between documents and terms. The results of experiments show that the recall and precision rates of Chinese news filtering using the character-based ap-proach incorporating the LSI technique are satisfactory.

1. Introduction

The rapid growth of the Internet has created the need of the Network Information Retrieval Systems. Most of the famous systems that assist people in locating information on the Internet, such as Lycos, Infoseek, Alta Vista, and WebWatcher [Armstrong 95], are designed for English information retrieval. To our knowledge, only the Csmart [Chien96] and GAIS [http://gais.cs.ccu.edu.tw/] systems are designed for Chinese information retrieval. However, information filtering is conceptually different from information retrieval, so we have to modify the techniques of information retrieval for

*Department of Computer Science, National Tsing Hua University, Hsin-Chu 30043 Taiwan R.O.C.
e-mail: shwu@cs.nthu.edu.tw, pcyang@cs.nthu.edu.tw, soo@cs.nthu.edu.tw

information filtering. In this paper, we assess the LSI technique for a hierarchical Chinese information filtering scheme. In particular, we assess the SVD approach to Chinese news filtering; to our knowledge, the SVD approach has never been investigated for the Chinese language.

Usenet news is one of the richest information resources on the Internet. Finding useful news among thousands of available news items is a crucial problem [Lang95]. Imagine a client user who needs a software agent to automatically recommend interesting news in Chinese from the Internet. Since news is updated every day, the traditional technique for information retrieval of using a fixed database will not work. Also, the task that a news filtering agent faces is to select relevant news, according to the user's interest or preference from a huge amount of dynamically growing news. Belkin and Croft [Belkin 92] pointed out that one major difference between information retrieval and filtering is that the queries in information retrieval typically represent the user's short-term interests while the user profiles in information filtering tend to represent the user's long-term interests. To model the user's long term interest, a user profile plays an important role in information retrieval [Mayeng 90] and filtering. Profiles can be represented in many ways and at different psychological and abstract levels. A collection of documents in a user's personal digital library may approximate the user profile. Information filtering is a document-find-document style of information retrieval. A document that is similar to the documents in the user's personal digital library is regarded as being relevant.

We adopt the vector space model [Yan 94] in our design of Chinese news filtering agents. In this model, each document is represented as a vector of weights of terms. We form each user profile by merging document vectors of the same interest category. The similarity of the incoming document vectors with the profile vectors can be computed by the means of cosine angles between the two vectors to determine if a document is to be filtered out.

In Chinese, there are no word delimiters to indicate the word boundaries; therefore, word segmentation is a difficult task. Many proper nouns or unknown words can not be found even in a word dictionary with a large vocabulary [Chien 95]. The number of different Chine-se characters is about 13,000, among which about 5,000 characters are the most commonly used characters. However, the number of Chinese words in a document collection set can easily be up to 1,000,000. To represent a personal profile in terms of words, the difficulty of word segmentation in Chinese must be dealt with [Chien 96]. We will show through experi-ments that without word segmentation, characterbased filtering incorporating LSI can be a satisfactory information filtering method.

The filtering method incorporating with LSI selects relevant documents whose contents have no exactly matched keywords. This is quite different from traditional techniques, such as the Boolean models. The probabilistic model, the Bayesian Belief Network Model [Turtle 91] [Ribeiro 96] shares similar feature. The Boolean models exactly match the document's terms with the combination of the search terms specified in the query. The probabilistic models estimate the degree of relevance between documents and a user query by considering the appearance frequency of certain terms in the documents and the user query, together with information about term distribution in the document collection.

Since individual terms and keywords are not adequate discriminators of the semantic content of documents and queries, the performance of the conventional retrieval models often suffers due either to missing relevant documents which are not indexed by the keywords specified in the query, or to retrieval of irrelevant documents which are indexed based on an unintended sense of the keywords in the query. Therefore, there has been great interest in text retrieval research that is based on semantics matching instead of strictly keyword matching.

Latent Semantic Indexing (LSI) using Singular-value Decomposition (SVD) is one approach to overcoming this deficiency of exact keyword matching techniques. We use truncated SVD to capture the semantic structure of the word usage in certain documents and hope that this usage can be applied to other documents. Using the singular value matrix from the truncated SVD, a high-dimensional vector space representing a term-document matrix is mapped to a lower dimension matrix that reflects the major concept factors in the specified documents while ignoring the less important ones. Terms occurring in similar documents will be nearer in the reduced vector space. With LSI, documents may satisfy a user's query when they share terms that are closer to each other in the reduced space. Since the reduced vector spaces are more robust indicators of the semantic meaning than are individual words, the performance may be better than that of the original space.

Several papers have reported the use of the LSI method. An example was assigning submitted manuscripts to the reviewers of the Hypertex'91 conference based on the interests of each reviewer; using the LSI method, a set of relevant manuscripts was sent to the reviewer [Dumais 92]. The automated assignment method achieved good matching between the reviewers and their interests just as did assignment by the human experts. Syu presented the technique of incorporating LSI into a neural network model for text retrieval [Syu96]. The performance, in terms of precision and recall, was comparable to that of the text retrieval model.

The remainder of this paper is organized as follows: Section 2 provides an overview of the Latent Semantic Indexing method applied to information retrieval, and how truncated SVD can be used as an LSI approach. Section 3 briefly reviews our information filtering scheme. Section 4 reports experimental results obtained using the LSI-based model, and Sec-tion 5 offers the discussion and a conclusion.

2. Latent Semantic Indexing method applied to information retrieval

Latent Semantic Indexing (LSI) is an extension of the vector space retrieval method. We assume that there is some unknown "Latent" association in the pattern of terms or keywords used among documents [Dumais 92], and try to estimate this latent association. Singular-Value Decomposition (SVD) is a technique for eigenvector decomposition and factor analysis used in statistics [Cullum 85], and Latent Semantic Indexing (LSI) using SVD is one approach used to model the latent semantic relationship between the documents and the index terms. This approach performs singular-value decomposition on a term-by-document matrix, thus generating a reduced space with lower dimension. The similarity between two documents is calculated according to the index terms used in each of the documents that occurr in other documents. Using the LSI representation, the documents satisfy a user query when they share terms of similar semantic meaning in the reduced vector space. The dimension of the resulting vector space is much smaller than the number of exact index terms used in a document collection (e.g., from several thousands to 100 or 300 [Dumais 94]), and a filtering model using LSI can save time and memory.

2.1 Singular-Value Decomposition (SVD) and truncated SVD

SVD is a reliable tool for matrix factorization. For any matrix A, A^TA has nonnegative eigen-values. The nonnegative square roots of the eigenvalues of A^TA are called the singular values of A, and the number of non-zero singular values is equal to the rank of A, *rank (A)*. Assume that A is an m *by* n matrix and that $rank(A) = r$; the singular -value decomposition of A is de-fined as

$$A = U W V^T,$$

where the size of U is m *by* m, the size of V is n *by* n, and the size of W is m *by* n. Both U and V^T are orthogonal matrices, i.e., $UU^T = I_m$, and $VV^T = I_n$; W is a diagonal matrix consisting of the singular values of A: $\sigma_1, \sigma_2, \cdots, \sigma_r$. And the σ_j's are the singular values of A, $\sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_r \geqq 0$, and $\sigma_j = 0$ for $j \geqq r+1$.

To apply SVD as an LSI tool, a term-by-document matrix A must be constructed. Then we can use SVD to generate the optimal approximation of the document representation specified by the matrix *A*. Since the singular values in the matrix *W* are ordered from largest to smallest, the first largest *k* singular values may be kept, and the remaining smaller ones are set to zero. As a result, the representations of the matrices *U*, *V*, and *W* can be reduced to obtain a new diagonal matrix W_k by removing columns and rows which are zeros from *W*; a matrix U_k can be obtained by removing the *(k+1)*st to the *m*th columns from *U*; and a matrix V_k can be obtained removing the *(k+1)*st to the *n*th rows from *V*. The product of the resulting matrices is a matrix A_k which is an approximation of the matrix *A*, and rank $(A_k)=k$:

$$A_k = U_k W_k V_k^T,$$

The relation between matrices *A* and A_k is shown in **Figure 1**. The LSI method using SVD can be viewed as a technique for deriving a set of non-correlated indexing of factors (i.e., the singular vectors) which represent different concepts in the usage of words in the document collection. The documents and queries are then represented by the vectors of factor values instead of the individual index terms. Using the *k*-largest factors, it may be possible to capture the most important latent semantic relation between documents and index terms, and to avoid unintended sense in word usage.

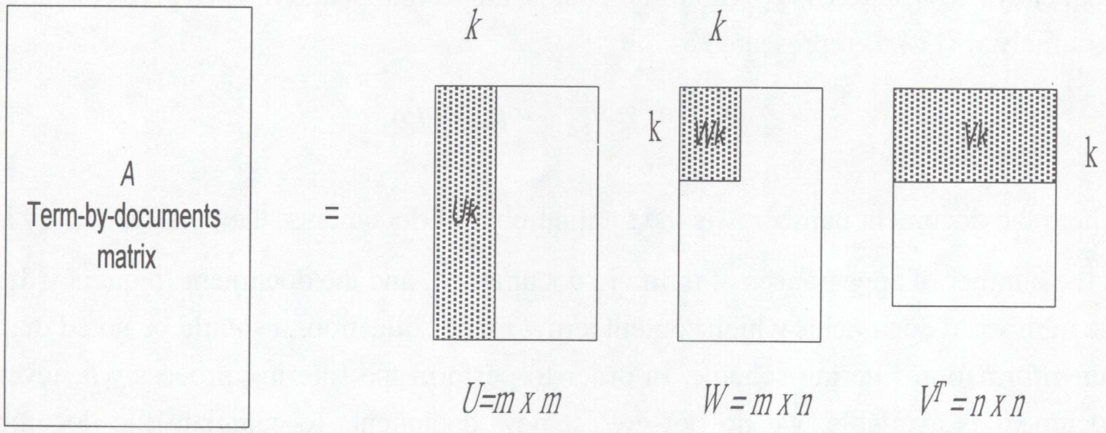


***Figure 1** An illustration of truncated SVD of a term-by-document matrix A.*

2.2 Document and query representations

Since the term-by-document matrix *A* has been reduced to a lower dimension matrix A_k, the vector which represents the query and all the incoming documents must be mapped to the same dimension of the matrix A_k. Using singular-value decomposition, a term-by-document matrix *A* is mapped into a reduced *k* by *n* matrix represented by $W_k V_k^T$, which relates *k* factors to n documents. A query *q*, originally of dimension size *m*, can be mapped into a size *k* vector *q'* :

$$q' = (q^T U_k W_k^{-1})^T.$$

The similarity between two documents is then computed using this shorter vector representation.

3. The character-based Chinese news filtering Scheme

3.1 The character-based vector representation of documents and personal profiles

A Chinese character is the basic processing unit and is used much like as the concept of a "term" in the IR denominating tradition; we use terms to refer to Chinese characters in this paper. In our approach, we need no stemming, no stop word list and no thesaurus. We represent the weight of a term in a given document by adopting Salton's well-tested *TFIDF* for-mula in IR, the term frequency (tf) multiplied by the inverse document frequency (idf) [Salton 89] [Salton 91]. That is, the weight of a term t in a given document d, namely w (t,d), is represented as

$$w\,(t,\,d) = tf_{t,d} * log\,(N/df_t),$$

where the document number *N* is the total number of documents, the term frequency $tf_{t,d}$ is the number of appearances of term *t* in document d, and the document frequency dft is the number of documents which content term t in the collection. It should be noted that in our information filtering scheme, in order to perform the filtering process whenever a document is available, we do not collect new documents to calculate the document frequency. Instead, we use a fixed set of document to represent the documents frequency for all the incoming documents.

A document *D* can be represented as a vector *V* with elements *v1, v2,* ⋯, *vn*, where n is equal to the size of the character vocabulary, and *vi* is the weight of term i in the document. All vectors are normalized for convenience and by convention. We can calculate the similarity between two documents *Di* and *Dj* by means of the cosine of the angle between their vector representations:

$$\text{Similarity } (Di, Dj) = \frac{Vi * Vj}{\| Vi \| \| Vj \|} \quad ,$$

where V_i is the vector representation of *Di*; * represents the inner product between two vectors; ||*Vi*|| represents the norm of a vector *Vi*. Based on the formula, two documents with the same character set will have the highest value of similarity between them because the inner product of the two document vectors will be one while two documents without any character in common will have the lowest similarity: zero.

We merge the document vectors in the same interest group (grouped either by the user or by a classification/clustering agent) into a higher level profile vector by summing up their vector. The profile vector is also normalized.

3.2 The tasks of a news filtering agent

A Chinese character-based news filtering agent will carry with it a set of weights in terms of inverse document frequencies as discussed above for a vocabulary of terms (characters), and a profile vector that represents a certain interest category and a similarity threshold associated with the profile vector. For each news document in the news server, the filtering agent will convert it first into a document vector; then, the similarity between the document vectors and the profile vectors will be computed according to the similarity measurement method discussed in section 3.1. If the similarity of the document with the profile is lower than the threshold, it is filtered out. In order to calculate every different recall rate, we treat the thresh-old as a variable in the experiment.

3.3 The hierarchical information filtering scheme

The hierarchical information filtering scheme [Soo 97] reduces the agent's total task. By com-posing profile vectors, we reduce the number of vectors that each agent must compare with document vectors on the Web. A user picks some interesting documents, and then the system converts the documents into vectors and merges the vectors into one profile vector. As the number of profile vectors increases (each representing a group of interest to the user), the lower-level profile vectors are combined to form higher-level profile vectors. We may assume that the final highest level profile vector can represent an overall interest of the user. The intelligent news filtering agent can then use this profile vector to search for relevant documents on the Web. All the documents that pass the higher-level filtering process are sent to the lower-level filter. Previous work showed that the hierarchical information filtering scheme can save computation time while maintaining the same recall-precision rate [Soo 97].

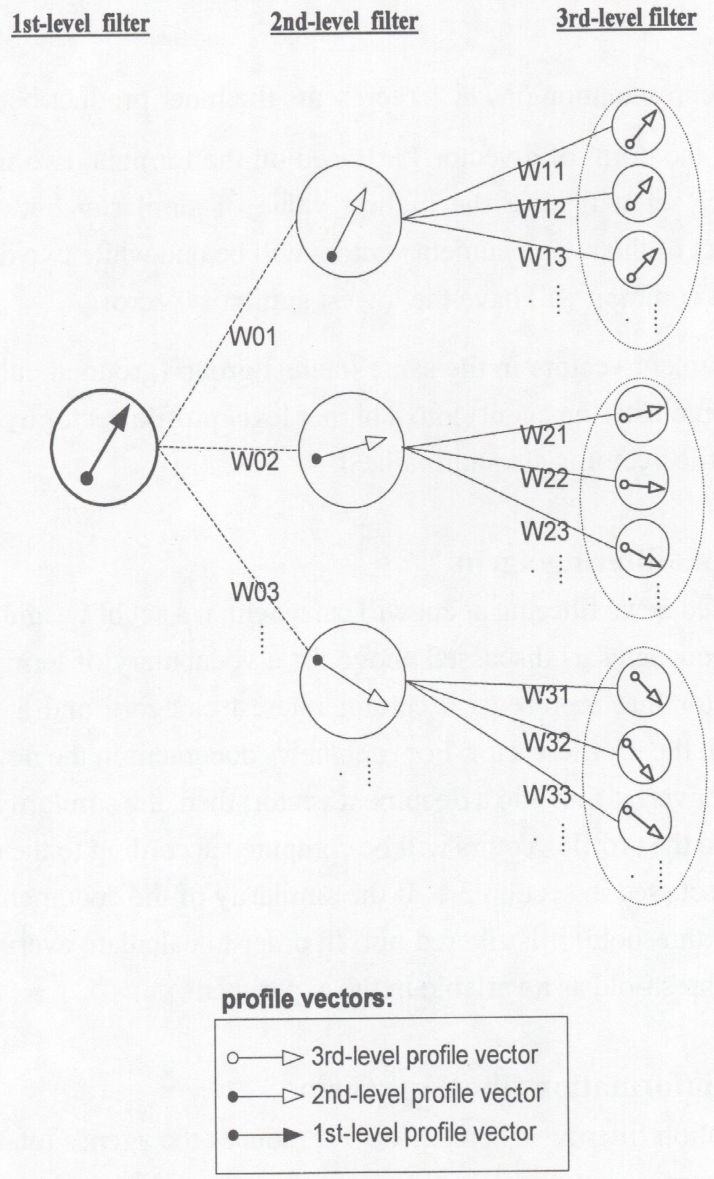


Figure 2 The hierarchical information filtering scheme.

4. Experimentation

Since the objective answer to the question of whether a document is relevant to the profile is not available, we tried to use the categories given by the news press to objectively form the groups of different interests to the user. Then, we assessed our method. The software package used to solve the SVD was MATLAB.

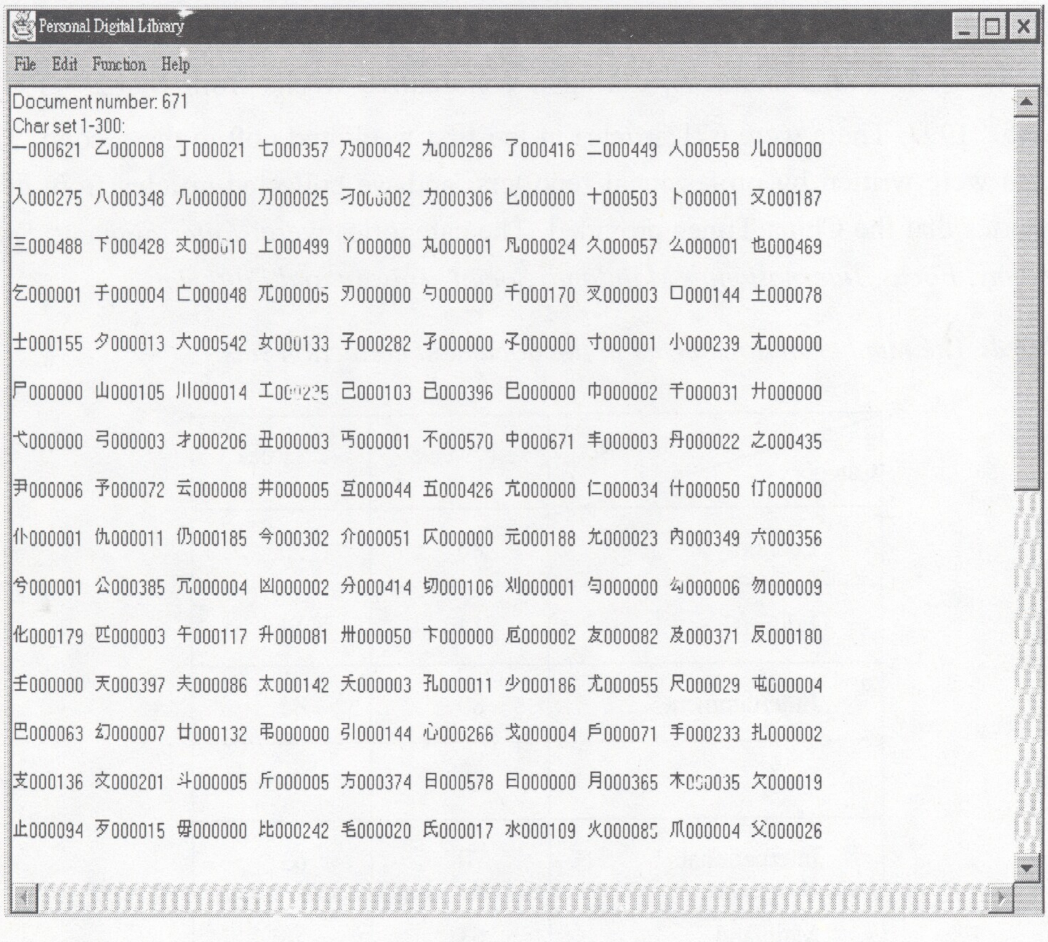
4.1 Data collection and document vectorization

We gathered three sets of articles from the on-line China Times [http://www.chinatimes.com.tw/] during 2 consecutive weeks from Mar.2nd 1997 to Mar.15th 1997. There were 671 articles in the first week and 669 in the second. These articles were written by professional reporters, and we collected articles from all the categories that the China Times provided. The categories were: *Entertainment, Sports, Economy, Focus, International, Mainland, Social, Taiwan and Editorial*.

Table 1. *The number of documents in the document collection sets.*

Set Category	1st week	2nd week
Economy	86	95
Editorial	14	14
Entertainment	80	82
Focus	80	79
International	70	63
Mainland	63	58
Social	67	73
Sports	90	87
Taiwan	114	111
Total	671	669

Table 1 shows the number of documents in each of the categories. The length of each article was about 500-2000 Chinese characters. In order to test wheather the frequency of words in the document collection sets was stable or not, we used the 671 articles in the first week as the training set to compute the document frequency *dft* for each term *t* (see **Figure 3**)and the 669 articles in the second week as the testing set for the filtering experiment.



```
Document number: 671
Char set 1-300:
一000621 乙000008 丁000021 七000357 乃000042 九000286 了000416 二000449 人000558 儿000000

入000275 八000348 几000000 刀000025 刁000002 力000306 匕000000 十000503 卜000001 又000187

三000488 下000428 丈000010 上000499 丫000000 丸000001 凡000024 久000057 么000001 也000469

乞000001 于000004 亡000048 兀000005 刃000000 勺000000 千000170 叉000003 口000144 土000078

士000155 夕000013 大000542 女000133 子000282 孑000000 孓000000 寸000001 小000239 尢000000

尸000000 山000105 川000014 工000225 己000103 已000396 巳000000 巾000002 干000031 廾000000

弋000000 弓000003 才000206 丑000003 丐000001 不000570 中000671 丰000003 丹000022 之000435

尹000006 予000072 云000008 井000005 互000044 五000426 亢000000 仁000034 什000050 仃000000

仆000001 仇000011 仍000185 今000302 介000051 仄000000 元000188 允000023 內000349 六000356

兮000001 公000385 冗000004 凶000002 分000414 切000106 刈000001 勻000000 勾000006 勿000009

化000179 匹000003 午000117 升000081 卅000050 卞000000 厄000002 友000082 及000371 反000180

壬000000 天000397 夫000086 太000142 夭000003 孔000011 少000186 尤000055 尺000029 屯000004

巴000063 幻000007 廿000132 弔000000 引000144 心000266 戈000004 戶000071 手000233 扎000002

支000136 文000201 斗000005 斤000005 方000374 日000578 曰000000 月000365 木000035 欠000019

止000094 歹000015 毋000000 比000242 毛000020 氏000017 水000109 火000085 爪000004 父000026
```

***Figure 3** A partial view of the document frequency dft for each term t.*

The articles were first transformed into normalized document vectors as discussed in section 2.1; all English characters and Arabic numerals were ignored. The similarity between two documents was then equal to the inner product of two vectors. **Figure 4** shows the similarity graph of the testing data; each document compared to all six hundred documents . The six hundred documents were selected from the document collection and ordered from 1 to 600 according to the categories: Economy, Editorial, Entertainment, Focus, International, Mainland, Social, Sports, and Taiwan, respectively; as in **Table 1**.

The gray levels represent the similarity values between two documents; the darkest point is of the highest similarity value. We can see that documents in the same category tended to have higher similarity values. This can be inferred from the fact that in the similarity graph, the darker points seem to form a symmetrical region along the diagonal line.

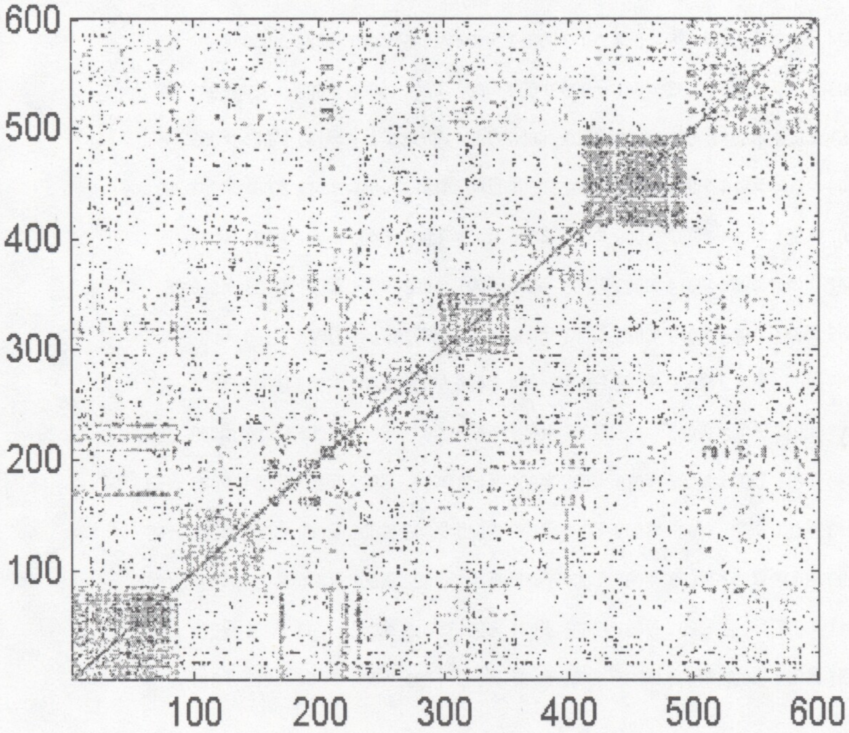
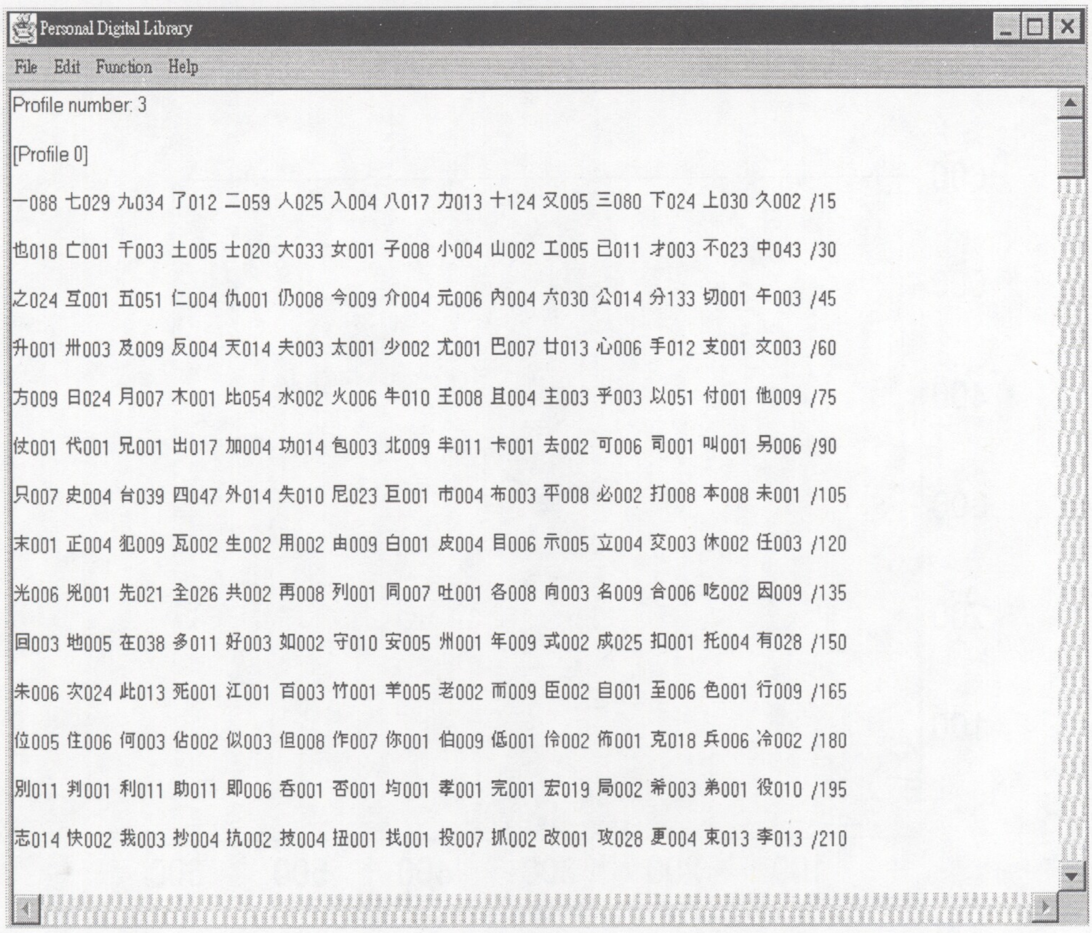


Figure 4 *The similarity graph of the testing data. Each document compared to all six hundred documents.*

4.2 Experiments on information filtering with SVD

To mimic a user's interests, we randomly choose news articles from three categories (Entertainment, Sports, and Economy) on the same day (Mar. 2nd) to form the initial user's profiles. A user profile can be treated as a set of documents which are transformed into normalized document vectors as shown in **Figure 5**.



```
Profile number: 3

[Profile 0]

一088 七029 九034 了012 二059 人025 入004 八017 力013 十124 叉005 三080 下024 上030 久002 /15

也018 亡001 千003 土005 士020 大033 女001 子008 小004 山002 工005 已011 才003 不023 中043 /30

之024 互001 五051 仁004 仇001 仍008 今009 介004 元006 內004 六030 公014 分133 切001 午003 /45

升001 卅003 及009 反004 天014 夫003 太001 少002 尤001 巴007 廿013 心006 手012 支001 文003 /60

方009 日024 月007 木001 比054 水002 火006 牛010 王008 且004 主003 乎003 以051 付001 他009 /75

仗001 代001 兄001 出017 加004 功014 包003 北009 半011 卡001 去002 可006 司001 叫001 另006 /90

只007 史004 台039 四047 外014 失010 尼023 巨001 市004 布003 平008 必002 打008 本008 未001 /105

末001 正004 犯009 瓦002 生002 用002 由009 白001 皮004 目006 示005 立004 交003 休002 任003 /120

光006 兇001 先021 全026 共002 再008 列001 同007 吐001 各008 向003 名009 合006 吃002 因009 /135

回003 地005 在038 多011 好003 如002 守010 安005 州001 年009 式002 成025 扣001 托004 有028 /150

朱006 次024 此013 死001 江001 百003 竹001 羊005 老002 而009 臣002 自001 至006 色001 行009 /165

位005 住006 何003 佔002 似003 但008 作007 你001 伯009 低001 伶002 佈001 克018 兵006 冷002 /180

別011 判001 利011 助011 即006 吞001 否001 均001 孝001 完001 宏019 局002 希003 弟001 役010 /195

志014 快002 我003 抄004 抗002 技004 扭001 找001 投007 抓002 改001 攻028 更004 束013 李013 /210
```

Figure 5 *A partial view of the profile vectors in the experiment.*

To evaluate the effectiveness of news filtering based on the character-based method for Chinese news document, the tf-idf weighting and vector space model were adopted in our experiment on the nine news categories, and we tried different k values when applying truncated SVD. As discussed above, in our hierarchical information filtering scheme, several arbitrary articles from each category in the training set were selected and merged into one profile document. The profile document was then transformed into a normalized vector, called a profile vector. By comparing the profile and document vectors in the test set, we could retrieve the most similar documents in the test set and measure the precision against different recall values as plotted in **Figures 6-8**.

From **Figures 6-8**, we observe that different values of *k* in SVD had different performance. The performance was worse either when the *k* value was small, e.g., 2, or when the *k* value was large, e.g., 100. The experiments showed taht the suitable value for *k* is 10 for our document collection sets. Dumais suggested that the probable *k* value is from 100 to 300 for an English document [Dumais 94]. The difference may come form the different language stuctures of Chinese and English. However, the result is what we want. The great reduction of the vector dimension saves a lot of memory space and time needed to further utilize the document vectors. We performed more experiments to justify our observation.

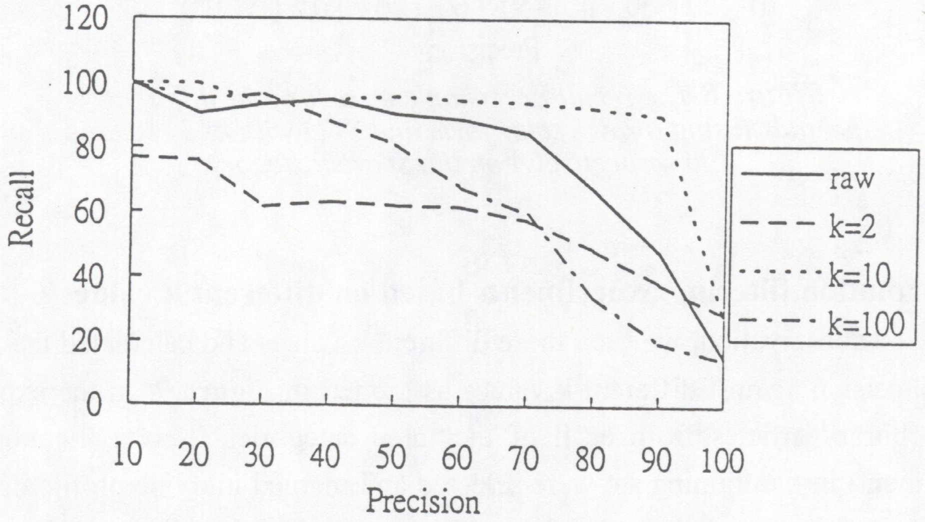


***Figure 6** The Recall-precision curves of four different methods (using a raw vector form and SVD with k=2, 10, and 100, respectively) on the Economy category.*

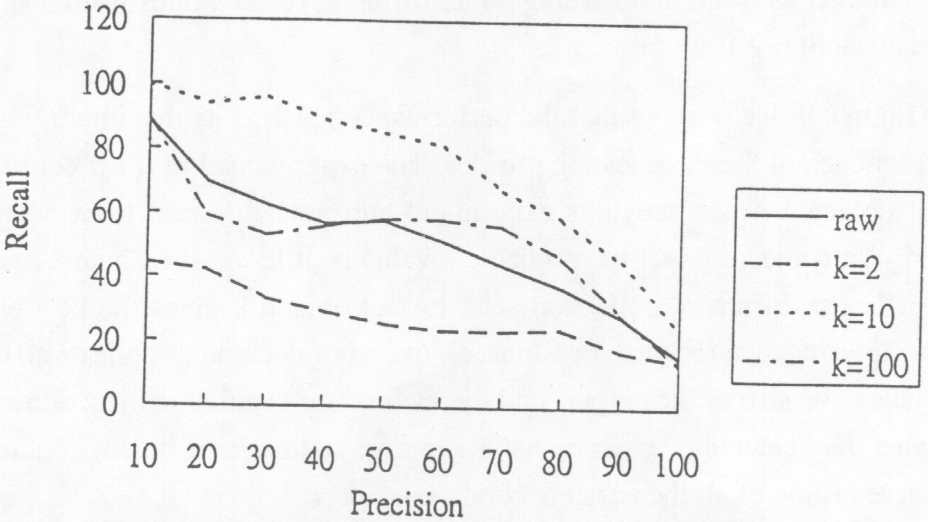


***Figure 7** The recall-precision curves for four different methods (using a raw vector form and SVD with k=2, 10, and 100, respectively) on the Entertainment category.*

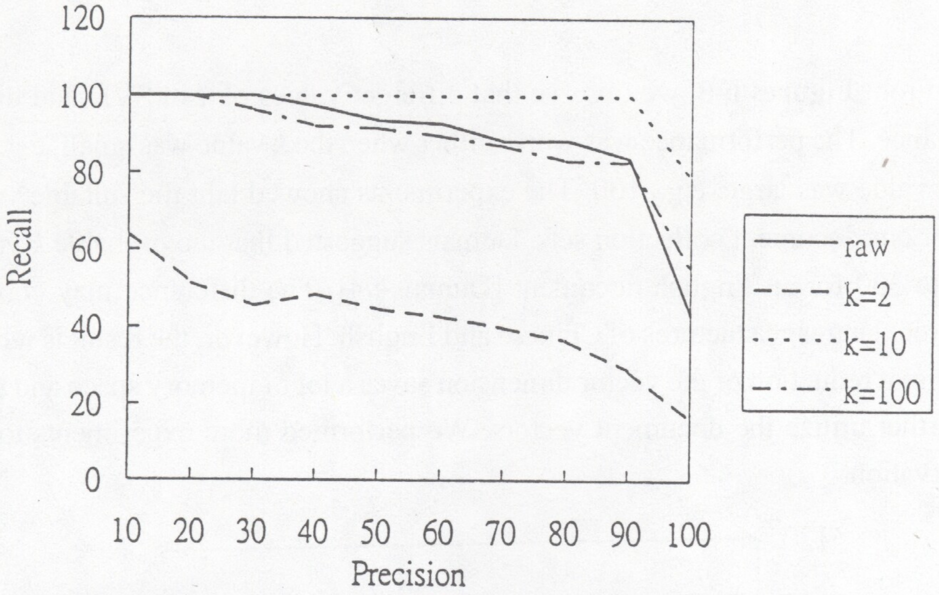


***Figure 8** The recall-precision curves for four different mehtods (using a raw vector form and SVD with k=2, 10, and 100 respectively) on the Sports category.*

4.3 Information filtering experiments based on different k values

To justify our observation, we tried more different k values and calculated the 11-point average precision against different k values as plotted in **Figure 9**. In the experiment, several arbitrary articles from each of the three categories (Sports, Economy, and Entertainment) in the training set were selected and merged into one profile document. The profile document was then transformed into a normalized vector named as a profile vector. By comparing the profile vector and document vectors in the test set, we could retrieve the most similar documents in the test set and measure the performance based on the 11-point average precision (average over different recall values from 0% to 100%, 10% in each step).

1. From **Figure 9**, we observe that the performance reached its maximum when k was about 10 for each of the three testing profiles. The experimental result is consistent with the result obtained in the previous experiment but quite different from what Dumais suggested. We conjecture that the probable *k* value is different for Chinese and English and for different document collection sets. In fact, which k gives the best result may depend on the language (English or Chinese), the set of documents collection, the length of the articles, the size of the corpus, and the method for evaluating precision and recall. The k value may change if experiments are carried out under different conditions. We only observed some of all the possible conditions.

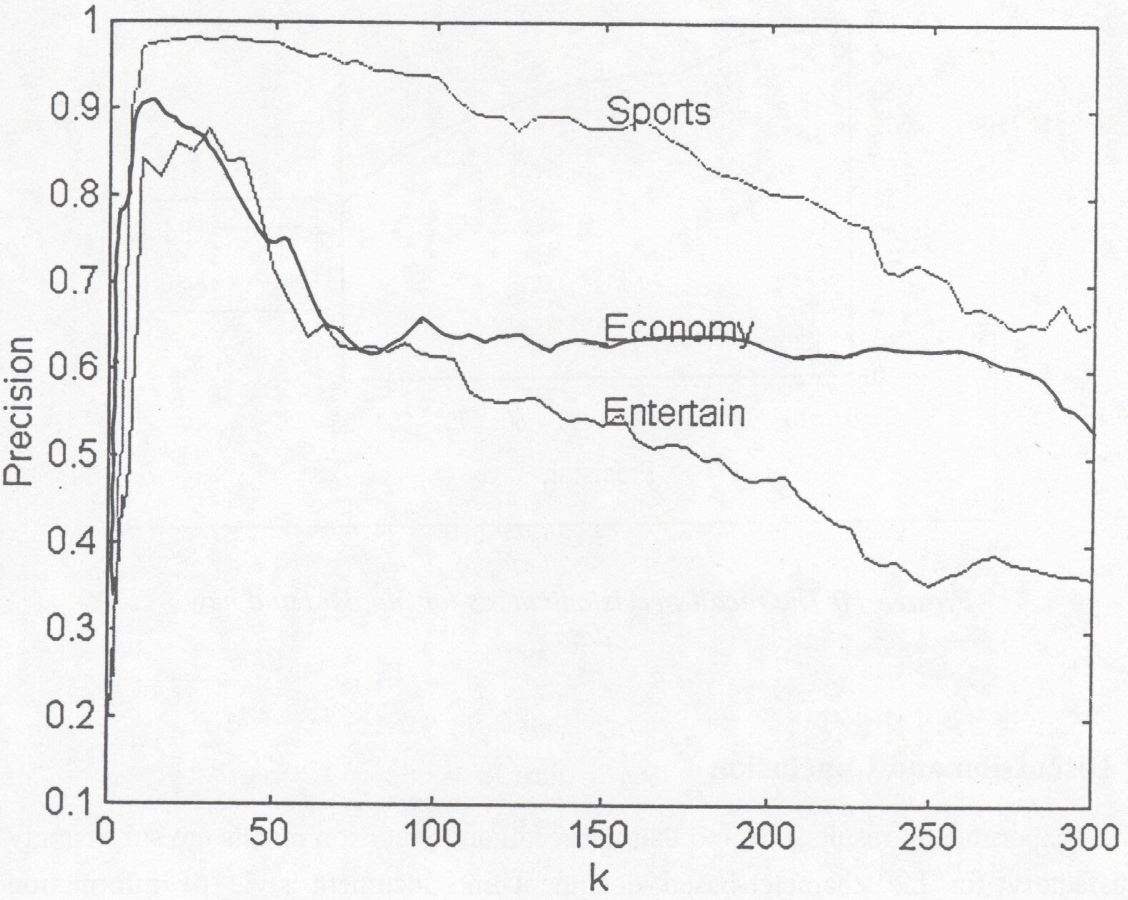


Figure 9 *Performance (11-points average precision) variation against different k*

4.4 Experiments on the testing data collected one year later

In order to find out if the filtering method is stable or not, we performed the same experiments one year later. We collected 1190 documents from the same WWW site. This time, we repeat the filtering process on the new testing data without changing our training data and profiles. The recall-precision performance decreased a little bit, namely, to 4.2% on average (see Figure 10).

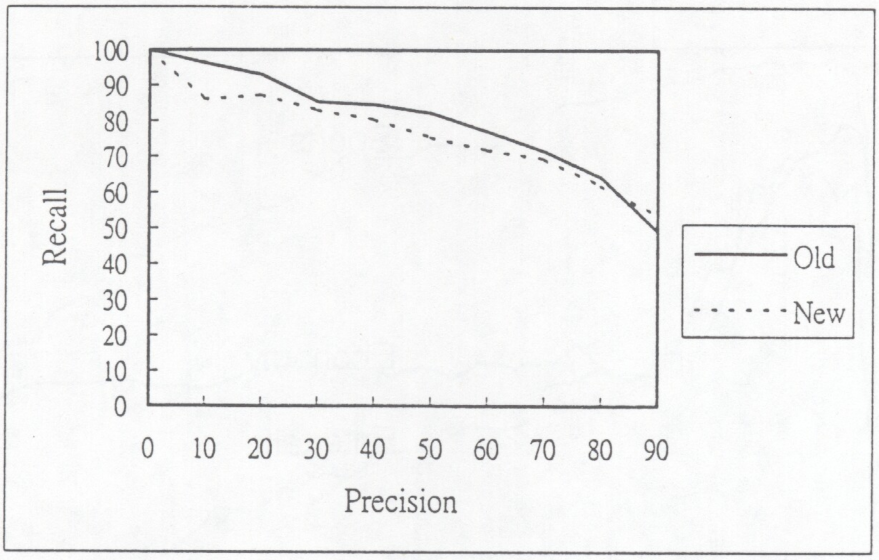


Figure 10 *The recall-precision curves for old data and new data*

5. Discussion and Conclusion

In the experimental results, we find that the recall and precision results are surprisingly satisfactory for the character-based document-find-document style of information filtering of Chinese news. The SVD technique can be used to reduce the storage space need of the term-by-document matrix and the processing time needed for further utilization. The difference in the performance for different choices of k with the truncated SVD value is quite interesting.

The results show that articles with similar character sets tend to have similar meaning, and that the semantic meaning of a Chinese news article can be, to some degree, implied by it's character set. Even though in Chinese, different orders of the same set of characters may have different meanings, and the same word may have ambiguities in parts of speech, character-based filtering can work well with the information provided by the context of an article.

The character-based information filtering scheme makes a lot of sense because no dictionary is rich enough that can contain all the possible words, and because the word segmentation task in Chinese is difficult. Only the weights and counts of the most commonly used characters in a documents collection set are needed to design an intelligent news filtering agent. A compressed matrix representation yields better performance and saves more computation time for the filtering agents. The SVD method can reduce the size of the term-by-document matrix and sort the significance of dimensions for the matrix. This is why a suitable choice of *k* will give better performance. The first *k* dimensions are necessary and sufficient to discriminate the categories. If we view stop words as noise, the larger the *k* value, the more noise will be considered. On the other hand, a small *k* may be insufficient for discrimination among categories. Several experiments in Chinese IR have shown that single-character based indexing cannot provide effective results when dealing with large size texts [Kwok 97]. However, in our domain, the news articles tended to be short.

Representing the user profile and performing news filtering hierarchically not only has the merit of reducing the computation cost, but also has potential for performing the information filtering task in a distributed and parallel manner. The efficiency will be improved even more if each profile vector runs independently on a distributed system. This could be achieved because of the independence property of the profile and document vectors; i.e., they do not interfere with each other when similar calculations are performed.

In the future, relevance feedback from the user can be used to improve the performance by adjusting several system parameters. It can be used to adjust the thresholds in each stage or to adjust the weights to combine lower level profile vectors into higher level ones. In this direction, we now are studying into machine learning techniques as neural networks [Pannu 95] [Syu 96].

Acknowledgment

This work was financially supported by the Institute for Information Industry and the National Science Council of Taiwan, Republic of China, under grant No. NSC86-2213-E-007-53.

References

Armstrong, R. and D. Freitag, T. Joachims, and T. Mitchell, "WebWatcher: A learning apprentice for the world wide web", *1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Stanford, March 1995.

Belkin, N.J. and Croft, W.B., "Information filtering and information retrieval: two sides of the same coin?", Comm. ACM 35, 12 (Dec.), pp. 29-38.

Chien, L.F., "Fast and quasi-natural language search for gigabytes of Chinese texts", In *Proceedings of 18th ACM SIGIR*, pp.112-120, 1995.

Chien, L.F., "An intelligent Chinese information retrieval system for the Internet", In *Proceedings of the ROCLING IX*, 1996.

Cullum, J.K. and R.A. Willoughby, "Lanczos Algorithms for Large Symmetric Eigen value

Computations - Vol. 1, Theory (Ch 5: Real Rectangular Matrices)", Brikhauser, Boston, 1985.

Dumais, S.T. and J. Nielsen, "Automating the Assignment of Submitted Manuscripts to Reviewers", In *Proceedings of the 15th ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp. 233-244, 1992.

Dumais, S.T., "Latent Semantic Indexing and TREC-2", *The Second Text Retrieval Conference (TREC-2)*, NIST Special Publication 500-215, pp. 105-115, 1994.

Kwok, K.L., "Comparing Representations in Chinese Information Retrieval", In *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp. 34-41, 1997.

Lang, K., "Newsweeder: learning to filter Netnews", *Proceedings of the Twelfth International Conference on Machine Learning*, 1995.

Mayeng, S. H. and R. R. Korfhage, "Integration of user profiles: models and experiments in information retrieval. Information Processing and Management", Vol. 26, No. 6, 1990.

Ribeiro, B.A.N. and R. Muntz, "A Belief Network Model for IR", In *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp.253-269, 1996.

Salton, G., "Automatic Text Processing", Addison Wesley, Reading, Massachusetts, 1989.

Salton, G., "Developments in automatic text retrieval", *Science 253*, 1991.

Pannu, A. S. and K. Sycara, "A learning personal agent for text filtering and notification", *Proceedings of the International Conference of Knowledge Based Systems (KBCS 96)*, Dec. 1996.

Soo, V.W., P.C. Yang, S. H. Wu and S.Y. Yang, "A Character-based Hierarchical Information Filtering Scheme for Chinese News Filtering Agents", *Proceedings of the Second International Workshop on Information Retrieval with Asian Languages (IRAL-97)*, Oct. 1997.

Syu, I., S. D. Lang, and N. Deo, "Incorporating latent semantic indexing into a neural network model for information retrieval", *Proceedings of the Fifth International Conference on Infor-mation and Knowledge Management*, Nov. 1996.

Turtle, H. and W. B. Croft, "Evaluation of an Inference Network based Retrieval Model", *ACM Transactions on Information Systems*, Vol. 9, No. 3, July 1991.

Yan, T. W. and H. Garcia-Molina, "Index structures for information filtering under the vector space model", *Technical Report STAN CS-TR-93-1494*, Nov. 1993.

Computational Linguistics and Chinese Language Processing
vol. 3, no. 2, August 1998, pp. 79-92

Noisy Channel Models for Corrupted Chinese Text Restoration and GB-to-Big5 Conversion

Chao-Huang Chang*

Abstract

In this article, we propose a noisy channel/information restoration model for error recovery problems in Chinese natural language processing. A language processing system is considered as an information restoration process executed through a noisy channel. By feeding a large-scale standard corpus C into a simulated noisy channel, we can obtain a noisy version of the corpus N. Using N as the input to the language processing system (i.e., the information restoration process), we can obtain the output results C'. After that, the automatic evaluation module compares the original corpus C and the output results C', and computes the performance index (i.e., accuracy) automatically. The proposed model has been applied to two common and important problems related to Chinese NLP for the Internet: corrupted Chinese text restoration and GB-to-BIG5 conversion. Sinica Corpora version 1.0 and 2.0 are used in the experiment. The results show that the proposed model is useful and practical.

1. Introduction

In this article, we present a noisy channel (Kernighan *et al.* 1990, Chen 1996) / information restoration model for automatic evaluation of error recovery systems in Chinese natural language processing. The proposed model has been applied to two common and important problems related to Chinese NLP for the Internet: corrupted Chinese text restoration (i.e., 8-th bit restoration of BIG-5 code through a non-8-bit-clean channel), and GB-BIG5 code conversion. The concept follows our previous work on bidirectional conversion (Chang 1992) and corpus-based adaptation for Chinese homophone disambiguation (Chang 1993, Chen and Lee 1995). Several standard Chinese corpora are available to the public, such as NUS's PH corpus (Guo and Lui 1992) and Academia Sinica's sinica corpus (Huang *et al.* 1995). These corpora can be used for objective evaluation of NLP systems. Sinica Corpora version 1.0 and 2.0 were used in the

*E000/CCL, Building 51, Industrial Technology Research Institute, Chutung, Hsinchu 31015, Taiwan, R.O.C. E-mail: changch@e0sun3.ccl.itri.org.tw

experiments. The results show that the proposed model is useful and practical.

The Internet and World Wide Web are very popular these days. However, computers and networks are not designed for coding huge numbers of Chinese ideographic characters since they originated in the western world. This situation has caused several serious problems in Chinese information processing on the Internet (Guo 1996). While the popular ASCII code is a seven-bit standard which can be easily encoded in a byte (eight bits), thousands of Chinese characters have to be encoded in at least two bytes. In this paper, we explore two error recovery problems for Chinese processing problems on the Internet: corrupted Chinese text restoration and GB-to-BIG5 conversion.

Mainland China and Taiwan use different styles of Chinese characters (simplified in Mainland China and traditional in Taiwan) and have also invented different standards for Chinese character coding. In order to fit different Chinese environments, more than one version of a web page is usually provided, one in English, and the other(s) in Chinese. Chinese versions of web pages are encoded in either BIG5 (Taiwan standard) or GB (Mainland China standard). Furthermore, the Unicode version will become popular in the near future.

BIG-5 code is one of the most popular Chinese character code sets used in computer networks. It is a double-byte coding; the high byte range is from (hexadecimal) A1 to FE, 8E to A0, and 81 to 8D; and the low byte range is from 40 to 7E, and from A1 to FE. The most and second most commonly used Chinese characters are encoded in A440 to C67E, and C940 to F9D5, respectively; the other ranges are for special symbols and used-defined characters. On the Chinese mainland, the most popular coding for simplified Chinese characters is the GB Code. It is also a double-byte coding; the high byte and low byte coding ranges are the same, (hexadecimal) A1 to FE.

In most international computer networks, electronic mail is transmitted through 7-bit channels (so called non-8-bit-clean). Thus, if messages coded in BIG5 are transmitted without further encoding (using tools like *uuencode*), the receiver side will only see some *random code* messages. In the literature, little work can be found on this problem. S.-K. Huang of NCTU (Hsinchu) designed a shareware program called Big5fix (Huang 1995), which is the only previous solution we can find for solving this problem. The input file for Big5fix is supposed to be a 7-bit file. Big5fix divides the input into regions of two types: an English Region and a Chinese Region. The characters in the Chinese region are reconstructed based on collected character unigrams, bigrams, trigrams and their occurrence counts. Huang estimated the reconstruction accuracy to be 90 percent (95% for the Chinese region and 80% for the English region). It is well known that shareware programs are provided free of charge for the general public. The accuracy rates are estimated without large-scale experiments. Our proposed corpus-based evaluation method based on information restoration can be used for this purpose if a large-scale standard corpus is available.

In addition to automatic evaluation of the accuracy rate of Big5fix, we will describe an intelligent 8-th bit reconstruction system, in which statistical language models are used for resolving ambiguities. (Note that there is no similar ambiguity in a pure GB text, in which both high bits of the two bytes are set. As one reviewer has pointed out, practical GB documents may be a mixture of ASCII text and GB codes. In that case, the 8-th bit reconstruction problem exists if the channel is not 8-bit clean. However, solving the problem will require a method of separating ASCII text from GB codes. This is actually beyond the scope of this study.)

In comparison, the GB-BIG5 conversion problem, that is, converting simplified characters to traditional characters, is well known and especially important nowadays since information flows rapidly back and forth across the strait and in a great volume. In addition to dictionaries in book form and manuals of traditional character-simplified character correspondences, many automatic conversion systems have been designed. Some of the shareware programs and products are as follows: the HC Hanzi Converter shareware, KanjiWeb (漢字通), NJStar (南極星), AsiaSurf (亞洲通), and UnionWin (亞洲心). However, the tools on the Internet commonly used are still one-to-one code converters. Therefore, we can easily find many annoying GB-BIG5 conversion errors in articles published in some newsgroups, such as alt.chinese.text.big5 or articles published in the BIG5 version of HuaXiaWenZai (華夏文摘). Some typical errors are: " 家里 (裡)", " 几 (幾) 個 ", " 技朮 (術)", " 標准 (準)", " 關系 (係)", " 計划 (劃)", " 采 (採) 用 ", and " 制 (製) 造 ". In the above examples, a string contains a two-character word (outside the parentheses) and a single-character correction (inside the parentheses). In addition to automatic evaluation performed by the HC converter and KanjiWeb, we will introduce a new intelligent GB-BIG5 converter. The statistical Chinese language models used in the new converter include the inter-word character bigram (IWCB) and the simulated-annealing clustered word-class bigram (Chang 1994, Chang and Chen 1993).

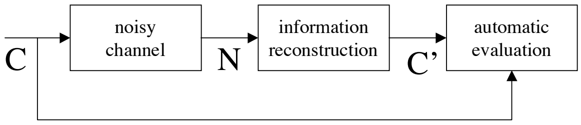


Figure 1: The proposed model.

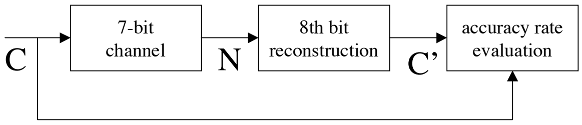


Figure 2: The proposed model for 8th bit reconstruction.

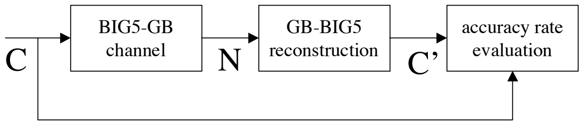


Figure 3: The proposed model for GB-BIG5 conversion.

2. Information Restoration Model for Automatic Evaluation

Extending the concepts of 'bi-directional conversion', the proposed corpus-based evaluation method applies the information restoration model for automatically evaluation of the performance of various natural language processing systems. As shown in Figure 1, a language processing system is considered to be an information restoration process executed through a noisy channel. By feeding a large-scale standard corpus C into a simulated noisy channel, we can obtain a noisy version of the corpus N. Using N as the input to the language processing system (i.e., the information restoration process), we can obtain the output results C'. After that, the automatic evaluation module compares the original corpus C and the output results C', and computes the performance index (i.e., accuracy) automatically.

The proposed evaluation model will obtain have near perfect results (obtain real performance) if the simulation of a noisy channel approaches to perfect. The perfect simulation would be one-to-one correspondence, or a process with near 100% accuracy. For example, for the syllable-to-character conversion system, a noisy channel, that is, character-to-syllable conversion, is not a one-to-one process (there are many PoYinZi, that is, homographs). However, it is not difficult to develop a character-to-syllable converter with accuracy higher than 98% (Chang 1992, Chen and Lee 1995). Thus, the proposed corpus-based evaluation method can be readily applied to estimate the conversion accuracy of a syllable-to-character conversion system. In fact, the proposed model can be applied to various types of language processing systems. Typical examples include *linguistic decoding* for speech recognition, word segmentation, part-of-speech tagging, OCR post-processing, machine translation, and two problems we will study in this article: 8-th bit reconstruction for BIG5 code and GB-to-BIG5 character code conversion. Indeed, the proposed model has its limitations. For problems where we can not perform nearly perfect noisy channel simulation, the performance (of either error recovery or evaluation) is inaccurate. Speech recognition may be one such problem (as one reviewer pointed out.)

Noisy channel simulation of the 8-th bit reconstruction process is perfect, i.e., one-to-one. The only thing the simulation needs to do is to set the 8-th bit of all bytes to zero. Thus, the proposed corpus-based evaluation method is ideal for application to this problem. The results will be completely correct. Figure 2 illustrates the proposed model for 8-th bit reconstruction for BIG5 code.

It is rather complex to simulate a noisy channel for the GB-BIG5 code conversion problem, not only because some traditional characters can be mapped to more than one simplified character (e.g., 乾 ⇨ 干、乾； 覆 ⇨ 复、覆), but also because even more characters can not be mapped to any suitable simplified characters. Nevertheless, the average accuracy rate for noisy channel simulation still approaches 100%, based on the occurrence frequency in large corpora. The proposed model is still applicable to this problem, as shown in Figure 3.

3. Preparation of Standard Corpora

In this study, we used the Academia Sinica Balanced Corpora, versions 1.0 (released 1995, 2 million words) and 2.0 (released 1996, 3.5 million words), to verify our proposed corpus-based evaluation model. Some statistics for the two corpora are listed in Table 1.

Table 1. *Academia Sinica Balanced Corpora, versions 1.0 and 2.0.*

Sinica Corpus	Size(bytes)	#files	#sentences	#words	#char.(inclu. symbols)	#char. (Hanzi only)
version 1.0	44,525,299	67	284,455	1,342,861	3,347,981	2,953,065
version 2.0	84,256,391	253	411,470	1,946,958	4,834,933	4,143,021

Word segmentation and sentence segmentation were used as originally provided by the Academia Sinica. The word segmentation follows the proposed standard set by ROCLING, which is an earlier version of the Segmentation Standard for Chinese Natural Language Processing (Draft). The part-of-speech tag set is a 46-tag subset simplified from the CKIP tag set (Huang *et al.* 1995). However, the word segmentations and part-of-speech tags were not used in our experiments. The following steps were used to restore the text using sentence segmentation:

(1) Grep (a Unix tool) was used to filter out the article classification headers, i.e., lines with leading %%; those sentence separator lines (lines filled with '*') were also removed.

(2) A small program called extract-word was used to extract the words in a sentence; part-of-speech information was removed. Output examples were something like " 我 起來 了 ，" ; " 太陽 也 起來 了 。"

(3) Words in a sentence into a character string, e.g., " 我起來了， ", and all files were concatenated into a single huge file.

(4) All user-defined special characters and non-BIG5 code were replaced with a special symbol ' □ '.

After pre-processing, the corpus became a single file, one sentence per line, and all the characters were double-byte BIG5 code. The statistics shown in Table 2 were calculated based on a pre-processed version of the corpora.

4. The 8-th Bit Reconstruction

4.1 System Design

The 8-th bit reconstruction (also called corrupted Chinese text restoration) problem has been described in Sections 1 and 2. We will not repeat the description here. To simulate a noisy channel, we simply set to zero the 8-th bit of each byte in the input. This could be done using a program of a few lines. We used Big5fix as a baseline system and developed an intelligent 8-th bit reconstruction system. The system resolves the ambiguity problem using statistical Chinese language models. The basic architecture follows our previous approach, called 'confusing set substitution and language model evaluation' (Chang 1994, 1996, Chang and Chen 1993, 1996). As shown in Figure 4, the characters in the input are replaced with corresponding confusing character sets, sentence by sentence. In this way, the number of sentence string candidates for an input sentence is generated. Then, the string candidates are evaluated using a corpus-based statistical language model. The candidate with the highest score (probability) is chosen to be the output of the system. Here, the 'confusing set substitution' step can be considered as inverse simulation of a 'noisy channel'.

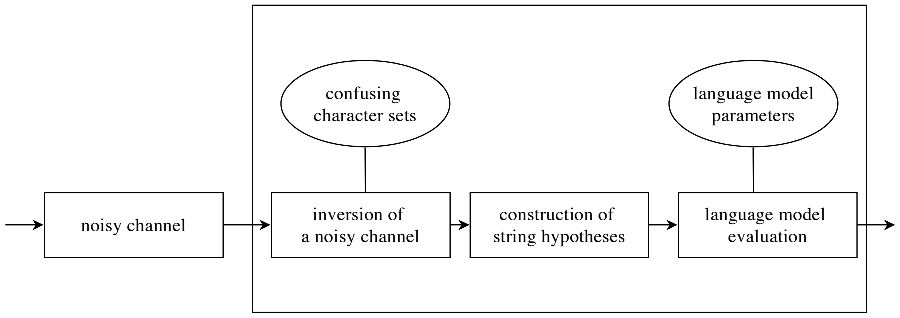

confusing character sets
language model parameters
noisy channel
inversion of a noisy channel
construction of string hypotheses
language model evaluation

Figure 4. The'confusing set substitution and language model evaluation' approach.

For the reconstruction problem, the 'confusing set' is very easy to set up. Since BIG5 is a double-byte code, we have at most two hypotheses for each character: the 8-th bits of all high-bytes are set to 1, and the 8-th bits of the low-bytes can be either 0 or 1 (depending on the code region). For example, the inverse simulation confusing set for 2440 (hex) contains two characters a440 「一」 and a4c0 「分」 , but the confusing set for 2421 (hex) only contains one character a4a1 「丑」 (a421 is outside of the coding region). In the system, we set up confusing sets for each of the 13,060 Chinese characters (including the 7 so-called Eten characters). Among them, 10,391 confusing sets contain two characters while the other 2,669 confusing sets contain only one character.

The statistical language model used in our system is an inter-word character bigram (IWCB) model (Chang 1993). The model is slightly modified from the word-lattice-based character bigram model of Lee *et al.* (1993). Basically, it approximates the effect of a word bigram by applying a character bigram to the boundary characters of adjacent words. The IWCB model is a variation of the *word-lattice-based Chinese character bigram* proposed by Lee *et al.* (1993). The path probability is computed as the product of the word probabilities and inter-word character bigram probabilities of the words in the path. For path H: $W_1 = W_{i_1 j_1}, ..., W_F = W_{i_F j_F}$, the path-probability estimated by the language model is

$$P_{LM}(H) = (\sum_{k=1}^{F} P(W_k))) \times \sum_{k=2}^{F} P(C_{i_k} | C_{j_{k-1}})$$

where Cik and Cjk are the first and last characters of the k-th word, respectively. This model is one of the best among the existing Chinese language models and has been successfully applied to Chinese homophone disambiguation and linguistic decoding. For details of the IWCB model, please refer to Lee *et al*. (1993) and Chang (1993).

4.2 Experimental Results

Table 2 compares the corpus-based evaluation results (the number of errors and the error rate %) of Big5fix and our intelligent 8-th bit reconstruction system (called CCL-fix).

Table 2. *Corpus-based evaluation results, Big5fix vs. CCL-fix.*

Sinica Corpus	Samples	#char.	Big5fix		CCL-fix	
Version 1.0	incl. symbols	3,347,981	125,915	3.76	57,862	1.72
	Hanzi	2,953,065	100,006	3.38	53,729	1.81
Version 2.0	incl. symbols	4,834,933	173,544	3.58	71,549	1.48
	Hanzi	4,143,021	111,809	2.69	70,758	1.70

As in Table 2 shows, the Hanzi reconstruction rates of Big5fix for the Sinica Corpora versions 1.0 and 2.0 are 96.62% and 97.31%, respectively. They are higher than the 95% rate estimated by Huang by 1.62%, 2.31%. The reconstruction rates of CCL-fix are 98.19% and 98.30%, respectively. This shows that the IWCB language model is indeed superior to the counts of character unigrams and bigrams. Note that the 1991 UD newspaper corpus (1991ud), consisting of more than seven million characters, was used to train the character bigrams in the IWCB model and the word bigrams used in simulated annealing word clustering. Some statistics for the **1991ud** corpus are as follows: 579,123 sentences, 7,312,979 characters, 4,761,120 word-tokens, and 60,585 word-types. The **1991ud** corpus is independent of the Sinica Corpus in both its publisher and sample date.

Table 4 lists the reconstruction error analysis results for the Sinica Corpus 1.0 obtained using the two systems. The table shows only the top 20 most frequent types of errors. Each entry shows the original character, the reconstructed character, and its occurrence count. For example, the most frequent error made by Big5fix is wrongly reconstructing ' 分 ' as ' 一 ', with 3,007 occurrences.

Table 3. *Reconstruction error analysis for the Sinica Corpus 1.0, Big5fix vs. CCL-fix.*

Rank	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Big5 fix	分一 3007	化了 1540	林者 1481	外全 893	全外 819	區記 797	色在 792	股松 771	來沒 734	省某 723	西多 722	價語 715	代用 712	反力 709	石加 676	吳找 672	十天 664	船爽 611	油迎 611	村困 601
CCL fix	一分 2298	了化 1388	分一 1375	又太 1327	沒來 1325	外全 1209	天十 1194	每並 887	多西 638	林者 577	十天 530	代用 491	象僅 484	某省 465	叫件 458	沙事 396	士方 386	女月 376	命所 359	吧扭 343

5. GB-to-Big5 Conversion

5.1 System Design

Three different simulations of the noisy channel for the GB-BIG5 conversion problem were performed in our experiments: we used (1) HC Hanzi Converter, version 1.2u, developed by Fung F. Lee and Ricky Yeung; (2) HC, revised version, in which the conversion table is slightly enhanced; and (3) the MultiCode of KanziWEB. These three systems all use the table-lookup conversion approach. Thus, the one-to-many mapping problem is not dealt with, and many errors can be found after converting GB code back to BIG5.

Table 4 lists the corpus-based evaluation results (the number of errors and the error rate %) for the three systems: HC1.2u, HC revised, and KanjiWEB .

Table 4. *Corpus-based evaluation results for HC1.2u, HC revised, and KanjiWEB.*

Sinica Corpus	Samples	# char.	HC1.2u		HC revised		KanjiWEB	
Version 1.0	incl. symbols	3,347,981	271,986	8.12%	46,162	1.37%	29,531	0.87%
	Hanzi	2,953,065	43,155	1.46%	43,070	1.45%	29,076	0.98%
Version 2.0	incl. symbols	4,834,933	403,954	8.35%	68,047	1.40%	43,705	0.90%
	Hanzi	4,143,021	60,113	1.45%	60,031	1.45%	40,561	0.98%

To deal with the one-to-many mapping problem in GB-BIG5 conversion, we have developed an intelligent language model conversion method which takes context into account. In the literature, Yang and Fu (1992) presented an intelligent system for conversion between Mainland Chinese text files and Taiwan Chinese text files. Their basic approach is to (1) build tables by means of classification; and (2) compute scores level by level. However, they resolve ambiguities by asking (the user), instead of using statistical language models. We take the 'confusing set substitution and language model evaluation' approach. The Chinese language models we use are (1) the IWCB model (introduced above) and (2) the SA-class bigram model. In the SA-class bigram model, the words in the dictionary are automatically separated into N_C word classes using a simulated-annealing word clustering procedure (Chang 1994, 1996, Chang and Chen 1993, 1996). The language models usually seek the optimal path in a word-lattice formed by candidate characters. The path probability of a word-lattice path is the product of lexical probabilities and contextual SA-class bigram probabilities. For a path of F words H = W_1, W_2, $\cdots$, W_F, the path-probability estimated by the language model is

$$P_{LM}(H) = (\sum_{i=1}^{F} P(W_i \mid \phi(W_i))) \times (\sum_{i=2}^{F} P(\phi(W_i) \mid \phi(W_{i-1})))$$

where $\phi(W_i)$ is the word class which W_i belongs to.

In the experiments, we used two versions of the SA-class bigram model, with N_C =200 and N_C =300, respectively. They will be denoted as the SA-200 and SA-300 models. The corpus for word clustering, **1991ud**, was first segmented automatically into sentences, and then into words by our Viterbi-based word identification program VSG (Chang and Chen 1993). The same lexicon and word hypothesizer were used in the language models.

To simulate the inverse noisy channel, we must set up confusing sets, that is, collections of variants and equivalent characters. In other words, it is a simulation of a one-to-many mapping from GB to BIG5. We found three sources of variants and equivalent characters: (1) the YiTiZi file in HC version 1.2u, (2) an annotation table of simplified characters in mainland China by Zang (1996), and (3) Appendix 10 in a project report (Hsiao *et al.*1993). Combining the three sources, we arranged four versions of confusing sets (A, B, C, and D), which were used and compared in the experiments. Some statistics of the four versions of confusing sets are shown in Table 5. The column label 'n-way' shows the number of BIG5 characters, each of which has *n* characters in its confusing set.

Table 5. *Statistics of the four versions of confusing sets.*

Confusing Set	Source	1-way	2-way	3-way	4-way	5-way
A	(1)	12644	364	48	4	0
B	(1)(2)	12397	597	57	9	0
C	(3)	12301	670	68	16	5
D	(1)(2)(3)	12144	777	117	15	7

5.2 Experimental Results

Table 6 compares the corpus-based evaluation results (the number of errors and the error rate %) of the three language models and four versions of confusing sets for GB-BIG5 conversion. (The input was provided by the Revised HC.)

Table 6. *Comparison of four versions of confusing sets with three language models.*

Sinica Corpus	Number of char.	IWCB				SA-200				SA-300			
		A	B	C	D	A	B	C	D	A	B	C	D
Version 1.0	2,953,065	12,742 0.43%	10,144 0.34%	12,997 0.43%	12,684 0.42%	15,574 0.52%	13,977 0.47%	16,867 0.57%	16,811 0.56%	13,614 0.44%	10,849 0.36%	13,500 0.45%	13,225 0.44%
Version 2.0	4,143,021	17,752 0.42%	14,139 0.34%	18,774 0.45%	18,465 0.44%	21,127 0.50%	18,593 0.44%	23,299 0.56%	23,297 0.56%	18,729 0.45%	15,439 0.37%	19,790 0.47%	19,554 0.47%
	468,609 (ambiguous)	17,752 3.78%	14,139 3.02%	18,774 4.01%	18,465 3.94%	21,127 4.51%	18,593 3.97%	23,299 4.97%	23,297 4.97%	18,729 3.99%	15,439 3.29%	19,790 4.22%	19,554 4.17%

We can see that the IWCB model achieved the best performance for the problem. The SA-300 model had comparative performance while the SA-200 model was relatively weak. However, we must note that the three intelligent conversion methods were all superior to KanjiWEB's one-to-one mapping method. The error rates are more than double those of the other methods in the one-to-one mapping system. Among the four versions of confusing sets, version B performed better than the others. Version C and version D had a larger set of confusing characters than version B, but their performance did not reflect this. The reason might have been that the larger sets make more unnecessary confusion. In contrast, Version A clearly had an insufficient number of confusing characters.

The evaluation did not exclude unambiguous characters. Among the 4,143,021 characters in the Sinica Corpus 2.0, 11.31% (468,609) were found to be ambiguous (316,889 2-way ambiguous, 125,297 3-way, 18,377 4-way, and 7866 5-way ambiguous). That is, a *random (or no-grammar)* language model had a 6.4% error rate. Evaluation of pure ambiguous characters revealed that the random model had an error rate of 55.96% while the best performance achieved by the models was 3.02%(IWCB), 3.29%(SA-300), 3.97%(SA-200), respectively.

Table 7 lists the conversion error analysis for by the four systems (HC1.2u, KanziWEB, IWCB, and SA-300) with confusing set version B. The notation is similar to that used in the above section. □ or blanks denote no corresponding character, a1bc(hex) or a140(hex).

Table 7. *Conversion error analysis for Sinica corpus 2.0 by the four systems*

Rank	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
HC 1.2u	裡里 6207	並并 5974	術朮 4574	幾几 3434	準准 2052	係系 1985	遊游 1866	劃划 1800	製制 1513	採采 1464	證証 1430	願愿 1321	臺台 1071	範□ 937	隻只 860	築□ 850	妳□ 825	豐丰 797	復複 758	衝沖 713
Kanzi WEB	裡里 6207	聽听 2922	係系 1985	遊游 1866	製制 1513	採采 1464	臺台 1071	妳奶 825	複復 781	衝沖 713	週周 668	牠它 667	症癥 620	蘇甦 603	幹干 564	儘盡 538	閒閑 455	碰踫 446	欸 440	佈布 439
WCB /B	臺台 885	妳你 825	台臺 761	牠它 603	欸□ 440	瞭了 383	佈布 367	昇升 325	裡里 319	週周 270	污汙 248	裏裡 220	周週 203	註注 196	夸誇 194	秘祕 183	佔占 181	儘盡 178	唸念 175	繫系 155
SA- 300B	裡里 1544	臺台 994	妳你 825	牠它 634	欸□ 440	秘祕 355	瞭了 353	佈布 310	佔占 263	註注 239	污汙 237	周週 234	台臺 223	念唸 221	週周 212	昇升 206	裏裡 202	升昇 196	夸誇 194	証證 154

6. Concluding Remarks

In this article, we have presented a corpus-based information restoration model for automatic evaluation of NLP systems and applied the proposed model to two common and important problems related to Chinese NLP for the Internet: 8-th bit restoration of BIG-5 code through a non-8-bit-clean channel and GB-BIG5 code conversion. The Sinica Corpora versions 1.0 and 2.0 were used in the experiment. The results show that the proposed model is useful and practical.

Acknowledgements

This paper is a partial result of project no. 3P11200 conducted by the ITRI under sponsorship of the Minister of Economic Affairs, R.O.C. Previous versions of the paper *in Chinese* appeared in JSCL-97 (Beijing) and Communications of COLIPS (Singapore). The original title of the paper presented at ROCLING XI was "Corpus-based Evaluation of Language Processing Systems Using Information Restoration Model". The current title was suggested by one of the ROCLING XI reviewers. Thanks are due to the reviewers for the constructive and helpful comments.

References

Chang, C.-H., Bidirectional Conversion between Mandarin Syllables and Chinese Characters. In *Proceedings of ICCPCOL-92*, Florida, USA, 1992 , pp. 174-181.

Chang, C.-H., Corpus-based Adaptation for Chinese Homophone Disambiguation. *Proceedings of Workshop on Very Large Corpora*, 1993, pp. 94-101.

Chang, C.-H. and C.-D. Chen, Automatic Clustering of Chinese Characters and Words. In *Proceedings of ROCLING VI*, Taiwan, 1993, pp.57-78.

Chang, C.-H. and C.-D. Chen, SEG-TAG: A Chinese word segmentation and part-of-speech tagging system. In *Proceedings of Natural Language Processing Pacific Rim Symposium (NLPRS '93)*, Fukuoka, Japan, 1993. pp. 319-327.

Chang, C.-H., Word Class Discovery for Contextual Post-processing of Chinese Handwriting Recognition. In *Proceedings of COLING-94*, Japan, 1994, pp. 1221-1225.

Chang, C.-H., Simulated Annealing Clustering of Chinese Words for Contextual Text Recognition, *Pattern Recognition Letters*, 17, 1996, pp.57-66.

Chang, C.-H. and C.-D. Chen, Application Issues of SA-class Bigram Language Models, *Computer Processing of Oriental Languages*, 10(1), 1996, pp.1-15.

Chen, H.-H. and Y.-S. Lee, An Adaptive Learning Algorithm for Task Adaptation in Chinese Homophone Disambiguation, *Computer Processing of Chinese and Oriental Languages*, 9(1), 1995, pp. 49-58.

Chen, S.-D., An OCR Post-Processing Method Based on Noisy Channel, Ph.D. Dissertation, National Tsing Hua University, Hsinchu, Taiwan, 1996.

Guo, J., On World Wide Web and its Internationalization. In the *COLIPS Internet Seminar Souvenir Magazine*, Singapore, 1996.

Guo, J. and H.-C. Lui, PH: a Chinese Corpus for Pinyin-Hanzi Transcription, TR93-112-0, Institute of Systems Science, National University of Singapore, 1992.

Hsiao J.-P. et al., Research Project Report on Common Chinese Information Terms Mapping and Computer Character Code Mapping across the Strait, 1993. (in Chinese)

Huang C.-R. et al. Introduction to Academia Sinica Balance Corpus, In *Proceedings of ROCLING VIII*, 1995, pp. 81-99. (in Chinese)

Huang,S.-K.,big5fix-0.10,1995.

ftp://ftp.nctu.edu.tw/Chinese/ifcss/software/unix/c-utils/big5fix-0.10.tar.gz

Kernighan, M.D., K.W. Church, and W.A. Gale, A Spelling Correction Program Based on a Noisy Channel Model. In *Proceedings of COLING-90*, 1990, pp. 205-210.

Lee L.-S. et al., Golden Mandarin (II) - an Improved Single-Chip Real-time Mandarin Dictation Machine for Chinese Language with Very Large Vocabulary. In *Proceedings of ICASSP-93*, II, 1993, pp. 503-506.

Yang, D. and L. Fu, An Intelligent Conversion System between Mainland Chinese Text Files and Taiwan Chinese Text Files, *Journal of Chinese Information Processing*, 6(2), 1992, pp.26-34. (in Chinese)

Zang, Y.-H., *How to Break the Barrier between Traditional and Simplified Characters*, China Times Culture, 1996. (in Chinese)

Computational Linguistics and Chinese Language Processing
vol. 3 no. 2, August 1998 pp. 93-114
©Computational Linguistics Society of R. O. C.

Statistical Analysis of Mandarin Acoustic Units and Automatic Extraction of Phonetically Rich Sentences Based Upon a Very Large Chinese Text Corpus

Hsin-min Wang*

Abstract

Automatic speech recognition by computers can provide humans with the most convenient method to communicate with computers. Because the Chinese language is not alphabetic and input of Chinese characters into computers is very difficult, Mandarin speech recognition is very highly desired. Recently, high performance speech recognition systems have begun to emerge from research institutes. However, it is believed that an adequate speech database for training acoustic models and evaluating performance is certainly critical for successful deployment of such systems in realistic operating environments. Thus, designing a set of phonetically rich sentences to be used in efficiently training and evaluating a speech recognition system has become very important. This paper first presents statistical analysis of various Mandarin acoustic units based upon a very large Chinese text corpus collected from daily newspapers and then presents an algorithm to automatically extract phonetically rich sentences from the text corpus to be used in training and evaluating a Mandarin speech recognition system.

Keywords: Mandarin speech recognition, statistical analysis of acoustic units, phonetically rich sentences, speech database

1. Introduction

Automatic speech recognition by computers can provide the most natural and efficient method of communication between humans and computers. Over the past decades, researchers all over the world have been involved in projects that have aimed to develop automatic speech recognizers, and many high performance systems have begun to emerge from research institutes and laboratories. However, experience has shown that the deployment of such speech recognition systems in realistic operating environments will

*Institute of Infromation Science, Academia Sinica, Taipei, Taiwan, R. O. C.
E-mail:whm@iis.sinica.edu.tw

require much better speech data to help us model the inherent variability in speech signals among different speakers and in different environments, and to help us evaluate performance under near realistic conditions. Thus, researchers all over the world have also participated in many efforts devoted to collecting speech databases of their own languages [Akira *et al*., 1990; Yu and Liu, 1990; Zue *et al*., 1990; Tseng, 1995; Wang, 1997], in addition to developing robust algorithms for speech recognition.

Because the Chinese language is not alphabetic and keyboard input of Chinese characters into computers requires a considerable amount of effort and training, Mandarin speech recognition is highly desired especially in the Chinese community. In Taiwan, speech recognition systems have been developed for a wide variety of applications, such as small to large vocabulary keyword spotting [Huang, Wang, and Soong, 1994; Bai, Tseng, and Lee, 1997], medium size vocabulary isolated word recognition for voice command and control [Chang *et al*., 1996], large vocabulary speech dictation [Lee *et al*., 1993a; Lee *et al*., 1993b; Huang and Wang, 1994; Lyu and Lee *et al*., 1995; Wang and Lee *et al*., 1995; Shen, 1996], limited-domain speech understanding [Lin, Wang, and Lee, 1997], and so on. An adequate speech database is certainly critical for successful development of such a system. Recently, the research teams who developed the Golden Mandarin series have designed sets of phonetically balanced training sentences based on different acoustic criteria by considering the statistical distributions of different acoustic units, and these sentences have been shown to be very effective when new users utter them according to a prompt on the computer screen to train their own dictation systems [Shen, 1996]. In this paper, we will first present statistical analysis of various Mandarin acoustic units based upon a very large Chinese text corpus collected from daily newspapers and then present an algorithm to automatically extract phonetically rich sentences from this text corpus to be used in efficiently training and evaluating a Mandarin speech recognition system.

The rest of this paper is organized as follows. The characteristic structure of the Chinese language is briefly introduced in section 2, and statistical analysis of various Mandarin acoustic units based upon a very large Chinese text corpus is discussed in section 3. The basic principles and a detailed description of the two-stage algorithm for automatic extraction of phonetically rich sentences from a text corpus are given in section 4 while two example experiments for extracting phonetically rich Chinese sentences are discussed in section 5. Finally, a few concluding remarks are given in section 6.

2. Characteristic Structure of the Chinese Language

In Mandarin Chinese, the total number of Chinese characters is believed to be unknown, but more than 10,000 characters are commonly used. A Chinese word is composed of from one to several characters, and combinations of these characters in fact give an almost unlimited number of Chinese words, among which at least some 100,000 of them are commonly used. All the Chinese characters are monosyllabic, and the total number of phonologically allowed syllables is only about 1345. Although the majority of Chinese words are composed of two or more syllables or characters, most of the characters can also be considered as monosyllabic words. This is why accurate recognition of all 1345 Mandarin syllables is believed to be the first key problem in Mandarin speech recognition with a very large vocabulary, and this is also why syllables are often chosen as the basic recognition target, very similar to the words used in systems for other alphabetic languages [Lee, Hon, and Reddy, 1990; Ney *et al*., 1994]. Of course, this small number of syllables also implies that a large number of homonym characters share the same syllable, and that there is a high degree of ambiguity. For example, on average, every syllable is shared by about 7-8 (10,000/1345) possible homonym characters. This one-to-many mapping relation from syllables to characters is certainly another key issue in Mandarin speech recognition with a very large vocabulary, and some relevant problems have been discussed in many papers [Lee *et al*., 1993a; Lee *et al*., 1993b; Lyu and Lee *et al*., 1995; Wang and Lee *et al*., 1995; Shen, 1996].

Tonal syllable (1345)				
Base syllable (416)				Tone (5)
INITIAL (22)	FINAL(41)			
	Medial (3)	Nucleus (9)	Ending (5)	

Table 1. *The phonological hierarchy of Mandarin syllables, where the number inside each bracket indicates the total number of units of that kind in Mandarin Chinese.*

Another very important feature of Mandarin Chinese is the existence of tones for syllables. Mandarin Chinese is a tonal language, in which each syllable is assigned a tone, and the tones have lexical meaning. There are basically a total of four lexical tones, i.e., the high-level tone (usually referred to as Tone 1), the mid-rising tone (Tone 2), the mid-falling-rising tone (Tone 3), and the high-falling tone (Tone 4) as well as one neutral tone (Tone 5). It has been found that the vocal tract parameters for Mandarin speech are only slightly influenced by the tones, and that the tones can be separately recognized primarily using pitch contour information [Wang and Lee, 1994; Wang and Chen, 1994]. If the differences among the syllables caused by tones are disregarded, then only 416 base syllables (i.e., syllable structures independent of tones) instead of 1345 different tonal syllables are required to cover the pronunciation of Mandarin Chinese. As a result, every tonal syllable can be considered as a combination of two independent parts, a tone from the five possible choices and a base syllable from the 416 possible candidates disregarding tones. In many large vocabulary Mandarin speech recognition systems [Lee *et al*., 1993a; Lee *et al*., 1993b; Huang and Wang, 1994; Lyu and Lee *et al*., 1995; Wang and Lee *et al*., 1995; Shen, 1996], tones and base syllables are, thus, recognized separately.

	IPA	SPA
Stop(6)	[p] [t] [k] [p'] [t'] [k']	b, d, g, p, t, k
Affricate (6)	[ts] [ts] [t] [ts'] [t s '] [t ']	z, Z, j, c, C, <
Nasal (3)	[m] [n] [η]	m , n, N
Liquid (1)	[l]	l
Fricative (6)	[f] [s] [s] [ ] [x] [z]	f, s, S, T, h, R
Vowel (10)	[a] [o] [ ] [e] [i] [u] [y] [ ] [ ] [ ]	a, o, e, E, i, u, U, Y, y, r
Null phohe*(1)		#

*The null phone is used to represent the null Initial

Table 2(a) *33 PLUs of Mandarin Chinese, where both International Phonetic Alphabet (IPA) and Simplified Phonetic Alphabet (SPA) symbols are listed for reference.*

Conventionally, each of the 416 Mandarin base syllables mentioned above can be decomposed into an INITIAL/FINAL format very similar to the consonant/vowel relations in other languages. There exists a total of 22 INITIALs and 41 FINALs for the 416 Mandarin base syllables, in which the INITIAL is the initial consonant of the base syllable while the FINAL is the vowel or diphthong part of the base syllable but including an optional medial or nasal ending. On the other hand, just as in many other languages, these 63 (22+41) INITIAL/FINALs can also be further decomposed into even smaller acoustic units, for example, phone-like units (PLUs). It has been found that a total of 33 phone-like units (PLUs) is sufficient to transcribe the 416 Mandarin base syllables. The phonological hierarchy of a Mandarin syllable is shown in Table 1, where the relationships among the tonal syllables, base syllables and tones, INITIAL/FINALs, and PLUs are shown. The 33 PLUs of Mandarin Chinese with IPA (International Phonetic Alphabet) representations are listed in Table 2(a), in which the corresponding simplified symbols used in this research (SPA, Simplified Phonetic Alphabet) are also listed for reference. The 22 INITIALs and 41 FINALs are listed in Table 2(b) and (c), respectively, all in the Simplified Phonetic Alphabet (SPA) for simplicity. It can be found that an INITIAL is always a PLU while a FINAL may contain one, two, or three PLUs in general. That is, a Mandarin base syllable is composed of two to four PLUs. Table 3 lists all 416 base syllables, where the vertical scale lists all 41 FINALs and the horizontal all 22 INITIALs.

#	b	p	m	f	d	t	n	l	g	k
h	j	<	T	Z	C	S	R	z	c	s

Table 2(b) *22 INITIALs of Mandarin Chinese*

Group	Member
1	Y y
2	a ai au an aN
3	o ou
4	e en eN er
5	i ia iE iai iau iou iEn in iaN iN io
6	u ua uo uai uEi uan uen uaN ueN uoN
7	U UE Uan Un UN
8	E Ei

Table 2(c) *41 FINALs of Mandarin Chinese*

			INITIAL																					
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
			#	Z	C	S	R	z	c	s	g	k	h	j	<	T	d	t	n	l	b	p	m	f
FINAL	1	Y		1	2	3	4																	
	2	y						5	6	7														
	3	a	8	9	10	11		12	13	14	15	16	17				18	19	20	21	22	23	24	25
	4	o	26																	414				
	5	e	27	28	29	30	31	32	33	34	35	36	37				38	39	40	41			409	
	6	ai	42	43	44	45		46	47	48	49	50	51				52	53	54	55	56	57	58	
	7	E	59																					
	8	Ei	60	61		62		63		64	65		66				67		68	69	70	71	72	73
	9	au	74	75	76	77	78	79	80	81	82	83	84				85	86	87	88	89	90	91	
	10	ou	92	93	94	95	96	97	98	99	100	101	102				103	104	105	106		107	108	109
	11	en	110	111	112	113	114	115	116	117	118	119	120				121	122	123	124	125	126	127	128
	12	an	129	130	131	132	133	134	135	136	137	138	139				410		140		141	142	143	144
	13	aN	145	146	147	148	149	150	151	152	153	154	155				156	157	158	159	160	161	162	163
	14	eN	164	165	166	167	168	169	170	171	172	173	174				175	176	177	178	179	180	181	182
	15	i	183											184	185	186	187	188	189	190	191	192	193	
	16	u	194	195	196	197	198	199	200	201	202	203	204				205	206	207	208	209	210	211	212
	17	U	213											214	215	216			217	218				
	18	ia	219											220	221	222			412	223				
	19	iE	224											225	226	227	228	229	230	231	232	233	234	
	20	iai	235																					
	21	iau	236											237	238	239	240	241	242	243	244	245	246	
	22	iou	247											248	249	250	251		252	253			254	
	23	ian	255											256	257	258	259	260	261	262	263	264	265	
	24	in	266											267	268	269	411		270	271	272	273	274	
	25	iaN	275											276	277	278			279	280				
	26	iN	281											282	283	284	285	286	287	288	289	290	291	
	27	ua	292	293	294	295					296	297	298											
	28	uo	299	300	301	302	303	304	305	306	307	308	309				310	311	312	313	314	315	316	317
	29	uai	318	319	320	321					322	323	324											
	30	uEi	325	326	327	328	329	330	331	332	333	334	335				336	337						
	31	uan	338	339	340	341	342	343	344	345	346	347	348				349	350	351	352				
	32	uen	353	354	355	356	357	358	359	360	361	362	363				364	356	413	366				
	33	uaN	367	368	369	370					371	372	373											
	34	ueN	374																					
	35	uoN		375	376	377	378	379	380	381	382	383	384				385	386	387	388				
	36	UE	389											390	391	392			393	394				
	37	Uan	395											396	397	398				399				
	38	Un	400											401	402	403		415						
	39	UN	404											405	406	407								
	40	er	408																					
	41	io	416																					

Table 3 *416 Base Syllables of Mandarin Chinese*

3. Statistical Analysis of Mandarin Acoustic Units Based Upon A Very Large Chinese Text Corpus

The Chinese text corpus used here to analyze the statistical distribution of Mandarin acoustic units was collected from daily newspapers. English characters or other special symbols contained in the sentences were simply discarded; then, the remaining sentences were word identified [Chen and Liu 1992] and phonetic spelling indicated using a lexicon consisting of around 85,000 frequently used Chinese words [CKIP 1993]. All the words in the lexicon are composed of from one to four characters, and the number of words is analyzed in Table 4. Finally, the corpus consisting of a total of 22,660,835 sentences (271,360,277 characters or syllables) was used to analyze the statistical distribution of Mandarin acoustic units.

	1-character word	2-character word	3-character word	4-character word	Total
# of words	14052	48339	11559	10433	84383

Table 4 *The Number of Words Contained in the Chinese Lexicon used in this Research*

First, the analysis was based on the frequency counts of 416 base syllables in the corpus. The results are summarized in Table 5. It can be seen that the top 10 most frequently used base syllables cover more than 19% of base syllables used in everyday newspapers, the top 50 cover more than 50%, and the top 200 cover more than 92%. On the other hand, among the 416 base syllables, more than 100 of them have less than 1% frequency of occurrence, which means that around 25% of the base syllables are barely used in everyday newspapers. The top 30 most frequently used base syllables are listed in Table 6. Although most of the top 30 base syllables most frequently used at the beginning and end of sentences also belong to the overall top 30 most frequently used base syllables except that the order might be slightly different, it was found that some of them show very high frequency only at specific positions; e.g., the base syllables "ta", "tai", "dan", "Ze", and "biN" are very frequently used at the beginning of sentences while "Suo", "dian", and "Cu" are used at the end of sentences.

Group	No. of syllables in the group	Total frequency of occurrence (%)	Accumulated frequency (%)
1-10	10	19.4553	19.4553
11-30	20	18.5154	37.9707
31-50	20	12.3730	50.3437
51-100	50	22.3279	72.6716
101-150	50	12.6151	85.2867
151-200	50	7.2693	92.5560
201-300	100	6.5284	99.0844
301-416	116	0.9156	100.0000
total	416	100.0000	

Table 5 *The Frequency Counts of the 416 Base Syllables.*

	Beginning of sentences		Middle of sentences		End of sentences		Overall	
	base syllable	frequency of occurrence (%)	base syllable	frequency of occurrence (%)	base syllable	frequency of occurrence (%)	base syllable	frequency of occurrence (%)
1	ZuoN	4.3139	SY	3.8267	SY	6.5184	SY	3.9364
2	#i	3.8161	de	2.8260	#i	2.4045	#i	2.8137
3	SY	2.4485	#i	2.7542	Suo	2.3450	de	2.5035
4	ta	2.2782	ji	2.0613	dian	2.2258	ji	1.9116
5	jin	2.0577	ZY	1.6633	Cu	2.1890	ZY	1.6120
6	tai	2.0132	guo	1.5641	#Uan	2.1845	guo	1.4833
7	#iou	1.9962	#uEi	1.3024	li	1.8829	ZuoN	1.4783
8	zai	1.9309	#U	1.2790	huEi	1.7694	#uEi	1.3031
9	dan	1.9144	huEi	1.2252	ren	1.4151	li	1.2191
10	Ze	1.7445	ZuoN	1.2088	ZY	1.3987	huEi	1.1942
11	biN	1.7051	guoN	1.1543	#uEi	1.3348	#U	1.1927
12	#er	1.5319	li	1.1534	ZuoN	1.3311	#Uan	1.1244
13	ZY	1.3131	bu	1.1068	jian	1.2696	jin	1.1161
14	guo	1.3009	#Uan	1.0991	TiN	1.2681	#iou	1.0844
15	#uEi	1.2783	jin	1.0872	ji	1.2566	bu	1.0738
16	bu	1.2563	#u	1.0704	hou	1.2397	guoN	1.0621
17	li	1.2110	#iou	1.0572	de	1.0702	#u	1.0260
18	jiaN	1.2008	ren	1.0380	ti	1.0531	ren	1.0147
19	mEi	1.1709	ZeN	1.0132	hua	0.9774	zai	0.9552
20	<i	1.0862	zai	0.9341	Ti	0.9203	ZeN	0.9489
21	ji	1.0741	<i	0.9309	diN	0.8725	<i	0.9284
22	jiN	1.0522	da	0.8837	TiaN	0.8712	da	0.8547
23	#in	1.0296	jian	0.8064	#u	0.8646	jian	0.8266
24	#U	0.9548	fu	0.8021	guo	0.8597	#er	0.8040
25	da	0.9543	fa	0.7910	#an	0.8545	TiN	0.7724
26	#iE	0.9245	#er	0.7828	CaN	0.8387	fa	0.7652
27	duEi	0.8186	he	0.7624	ZaN	0.8233	tai	0.7629
28	guoN	0.8049	TiN	0.7385	jia	0.7903	fu	0.7482
29	ZeN	0.7585	sy	0.7300	Tian	0.7884	Cu	0.7450
30	#u	0.7446	jia	0.7151	fa	0.7813	jia	0.7097
total		46.6839		38.3677		44.3985		37.9707

Table 6 *The Frequency Counts of the Top 30 Most Frequently Used Base Syllables*

For speech recognition purposes, especially for continuous speech recognition, the co-articulation effects between adjacent syllables are usually significant, so recognition accuracy usually degrades clearly from isolated syllable recognition to continuous speech recognition. Many context-dependent acoustic modeling techniques which specially consider the contextual situation are, therefore, widely used to compensate for the co-articulation effects and, thus, improve recognition accuracy [Lee, Hon, and Reddy, 1990; Ney *et al*., 1994; Lyu and Lee *et al*., 1995; Wang and Lee *et al*., 1995]. Just as the frequency counts of the 416 base syllables are very different as discussed above, the concatenation combinations of base syllables are distributed over a wide range. Table 7 lists the frequency counts of tri-base syllables. Although there is a total of 71,991,296 (416^3) possible combinations of tri-base syllables, only 7,927,335 (11.01%) of them were found in the corpus. Among the existing tri-base syllable combinations, 2,671,395 of them were found only once while only 1,808,572 were found more than 10 times. Furthermore, it is worth noting that 11 of the tri-base syllable combinations appear more than 100,000 times in the corpus; they are "#uEi #Uan huEi" (委員會), "tai bEi SY" (台北市), "ZuoN hua min" (中華民), "hua min guo"(華民國), "TiN ZeN #Uan" (行政院), "li fa #Uan"(立法院), "bai fen ZY"(百分之), and so on. Since it is not feasible to collect sufficient speech data to include all the existing tri-base syllable combinations, even when only combinations which appear 10 or more times are considered, most Mandarin speech recognition systems are, in fact, based on sub-units, such as INITIAL/FINALs [Wang and Lee *et al.*, 1995; Chang *et al.*, 1996; Shen, 1996; Bai, Tseng, and Lee, 1997; Lin, Wang, and Lee, 1997].

Frequency counts	1	>1	>5	>10	>50	>100	>1000	>10000	>100000
# of tri-base syllables	2671395	5255940	2683364	1808572	586974	327852	28085	979	11
% of possible combinations	3.710719	7.300799	3.727345	2.512209	0.815340	0.455405	0.039012	0.001360	0.000015

Table 7: *The Frequency Counts of Tri-base Syllable*s.

Although there are 902 (22*41) possible INITIAL-FINAL combinations, only 416 of them are phonologically allowed, and they comprise the 416 base syllables of Mandarin Chinese. Furthermore, there are 9,152 (416*22) possible INITIAL-FINAL-INITIAL combinations (equal to the number of possible combinations of base syllables with the INITIALs of their following base syllables), among which 8,722 (95.30%) were found in the corpus. The most frequently used INITIAL-FINAL-INITIAL is "S-Y-#", which has 0.83% frequency of occurrence. The remaining combinations are distributed quite flatly such that, from the second most frequently used combination to the 100-th most frequently used combination, the frequency of occurrence decreases gradually from 0.42% to 0.12%. The top 100 most frequently used combinations cover 18.77% of the occurrence of all the INITIAL-FINAL-INITIAL combinations. On the other hand, there are 17,056 (416*41) possible FINAL-INITIAL-FINAL combinations (equal to the number of possible combinations of base syllables with the FINALs of their preceding base syllables), of which 14,321 (83.96%) were found in the corpus. Again, the distribution is quite flat such that, from the most frequently used combination to the 100-th most frequently used combination, the frequency of occurrence decreases gradually from 0.35% to 0.10% and the accumulated frequency of the top 100 most frequently used combinations reaches 15.75%. Table 8 lists the frequency counts of the INITIALs that appear at the beginning of sentences. It can be found that the top 6 most frequently used INITIALs, such as "#" (null INITIAL), "Z", "j", "d", "t", and "b", out of the 22 INITIALs account for more than

50% of the frequency of occurrence at the beginning of sentences. On the other hand, for the FINALs as the end of sentences, as shown in Table 9, the top 9 most frequently used FINALs, such as "i", "Y", "u", "iEn", "uo", "iN", "an", "uEi", and "e", out of the 41 FINALs also account for more than 50% of the frequency of occurrence in everyday newspapers.

INITIAL	#	Z	j	d	t	b	g	S	T	z	l
Frequency of occurrence (%)	16.9662	10.0083	9.0221	7.0685	6.0899	5.7940	4.7310	4.6953	4.6872	4.3168	4.0258
Accumulated frequency (%)	16.9662	26.9745	35.9966	43.0651	49.1550	54.9491	59.6801	64.3754	69.0626	73.3794	77.4053
INITIAL	m	h	<	C	R	f	s	n	c	k	p
Frequency of occurrence (%)	3.4716	3.0636	2.9544	2.1851	2.1048	2.0557	1.8689	1.5026	1.4073	1.2850	0.6957
Accumulated frequency (%)	80.8769	83.9405	86.8949	89.0801	91.1849	93.2405	95.1094	96.6120	98.0193	99.3043	100.00

Table 8 *The Frequency Counts of INITIALs at the Beginning of Sentences (in the Order of Frequency of Occurrence)*

FINAL	i	Y	u	iEn	uo	iN	an	uEi	e	uoN	else
Frequency of occurrence (%)	9.1773	8.5746	7.6424	6.8212	4.8508	4.5254	4.0051	3.9879	3.9704	3.4420	43.0029
Accumulated frequency (%)	9.1773	17.7519	25.3943	32.2155	37.0663	41.5917	45.5968	49.5847	53.5551	56.9971	100.00

Table 9 *The Frequency Counts of FINALs at the End of Sentences (in the Order of Frequency of Occurrence).*

	Beginning of sentences	Middle of sentences	End of sentences	Overall
Tone 1	26.4100	20.7775	18.3910	21.0327
Tone 2	25.1454	23.9060	22.0296	23.8369
Tone 3	17.0535	17.8385	13.9836	17.4352
Tone 4	31.3911	34.3228	44.2315	34.9039
Tone 5	0.0000	3.1552	1.3643	2.7913

Table 10 *The Frequency Counts of the 5 Tones*

Then, the analysis was based on the frequency counts of the 5 different tones in the corpus. The results are summarized in Table 10. It was found that these 5 tones are in the order of Tone 4, Tone 2, Tone 1, Tone 3, and Tone5, according to the frequency of occurrence, no matter whether they occur at the beginning, middle, or end of the sentences, except that Tone 1 is more frequently used than Tone 2 at the beginning of sentences. The frequency counts of Tone 5 should be smaller if phonetic labeling errors, such as " 請著 (Zuo2) 便服… " was phonetic spelling indicated as " 請著 (Ze5)...", " 求個

(ge4) 股表現… " as " 求個 (ge5) … ", " 能了 (liau3) 卻多年心願… " as " 能了 (le5) … ", and so on, which often occurred at the words or characters with more than one allowed pronunciation, were taken into account. Furthermore, Tone 5 syllables are barely used at the beginning of sentences, except for some exclamation sentences that contain only a single character, such as " 啊 (#a5)!", " 呀 (#ia5)!", " 嘿 (hEi5)!", " 哇 (#ua5)!", and so on. The total number of possible tri-tones is 125 (5^3). The frequency counts of these tri-tones according to the frequency of occurrence are shown in Figure 1 while the details for the top 20 most frequently used tri-tones are listed in Table 11. The accumulated frequency counts of the 5 tones according to 125 tri-tones (these 125 tri-tones are also in the order of frequency of occurrence) are further shown in Figure 2.

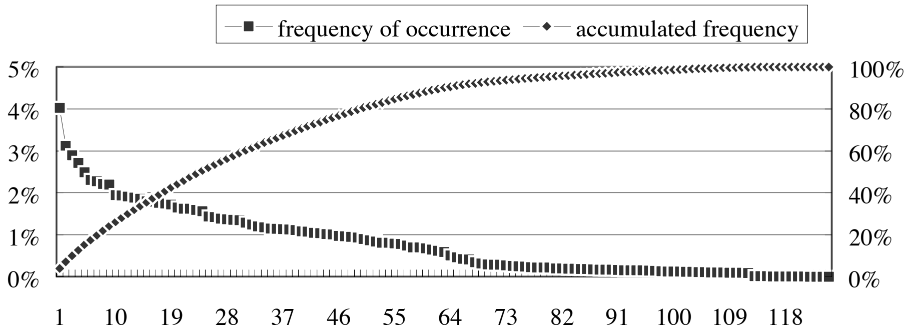


Figure 1 *The Frequency Counts of 125 Tri-tones in Their Order of Frequency of Occurrence*

Top	Tone combination	Frequency of occurrence (%)	Top	Tone combination	Frequency of occurrence (%)
1	4 4 4	4.0207	11	3 4 4	1.9423
2	2 4 4	3.1205	12	2 4 2	1.9109
3	4 2 4	2.8940	13	1 2 4	1.8858
4	4 4 2	2.7095	14	2 1 4	1.8579
5	1 4 4	2.4949	15	1 1 4	1.7971
6	4 4 3	2.3062	16	4 2 2	1.7906
7	4 1 4	2.2778	17	4 2 1	1.7653
8	4 3 4	2.2139	18	4 3 2	1.7466
9	4 4 1	2.1988	19	1 4 2	1.7214
10	2 2 4	1.9492	20	3 2 4	1.6506
total (top10)		26.1855	total (top20)		44.2540

Table 11 *The Frequency Counts of the Top 20 Most Frequently Used Tri-tones*

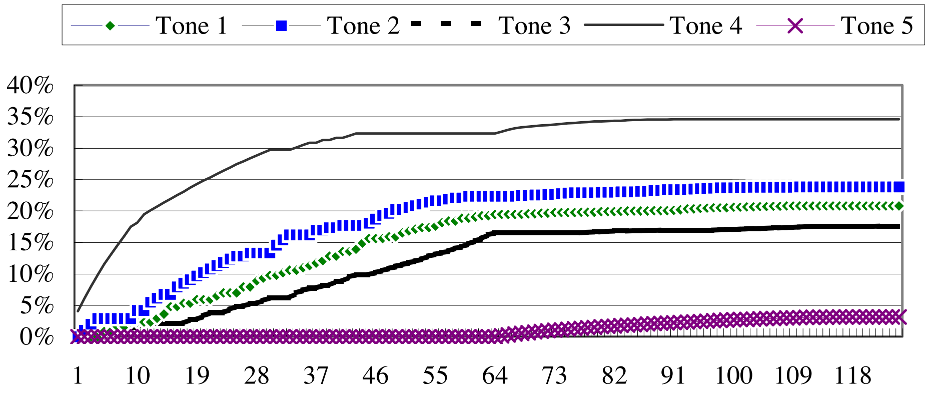


Figure 2 *The Accumulated Frequency Counts of the 5 Tones According to the 125 Tri-tones. (The 125 Tri-tones are shown in their order of frequency of occurrence.)*

Note that all the top 64 (4^3) most frequently used tri-tones are composed of 4 lexical tones, and that these tri-tones account for 90.62% of the frequency of occurrence while the rest are tri-tones with at least one Tone 5 syllable, and such tri-tones account for only 9.38% of the frequency of occurrence. Furthermore, among the tri-tones with at least one Tone 5 syllable, the 65-th to the 112-nd most frequently used tri-tones are tri-tones with only one Tone 5 syllable, the 113-rd to the 124-th are tri-tones with two Tone 5 syllables, and the 125-th (the least frequently used tri-tone) is a tri-tone composed of three Tone 5 syllables. The tri-Tone 5 combination has only 0.0061% frequency of occurrence; some examples are "zy5 men5 de5" (from 孩 " 子們的 " …), "men5 de5 le5" (from 看他 " 們的了 "), "men5 de5 ba5" (from 看我 " 們的吧 "), "ge5 ge5 de5" (from 一 " 個個的 "), etc. From Table 11, it is worth noting that all of the top 20 most frequently used tri-tones consist of at least one Tone 4 syllable. In fact, from Figure 2, we can further find that all of the top 30 most frequently used tri-tones consist of at least one Tone 4 syllable.

Though the above statistical analysis was performed based upon the text corpus collected from daily newspapers, in which the verbiage and the writing style might be slightly different from that of colloquial language in other specific domains, such as novels, magazines, and so on, it is believed that, except for some domain specific proper nouns, most of the frequently used words or characters are very similar across different domains. The statistical results obtained here based upon the text corpus collected from daily newspapers, therefore, provide valuable information which is certainly referable. Moreover, newspapers provide a reliable channel for collecting a very large-scale text corpus since they are generated day after day with the most up-to-date contents. This is the major reason why the statistical analysis was performed based upon a text corpus collected from daily newspapers in this study, and why, for many speech recognition systems, the language models are trained primarily based upon a text corpus collected from daily newspapers.

4. An Algorithm for Automatic Extraction of Phonetically Rich Sentences From a Text Corpus

For speech recognition purposes, so-called phonetically rich sentences consist of an almost smallest set of grammatically valid sentences, which not only include all necessary recognition units, but all these units should appear in some desired statistical distribution. Such a set of phonetically rich sentences will, then, be very useful in training and evaluating a speech recognition system. Because the recognition tasks (application domains, vocabulary, recognition units, such as phones, diphones, triphones, as well as other sub-word units, which are context-dependent or independent, etc.) are different for different recognition systems, trying to manually generate for each task such a set of phonetically rich sentences to be used in training and evaluating the system as was done in the past [Akira et al., 1990; Yu and Liu, 1990; Zue et al., 1990] will not be cost-effective. Furthermore, it's even very difficult for human experts to reproduce the statistical distribution of the recognition units in the recognition task while they are manually generating or selecting the training and testing sentences. Apparently, automatically generating such a phonetically rich sentence set from a text corpus which defines the task is highly desired. Here, a two-stage algorithm is, therefore, proposed.

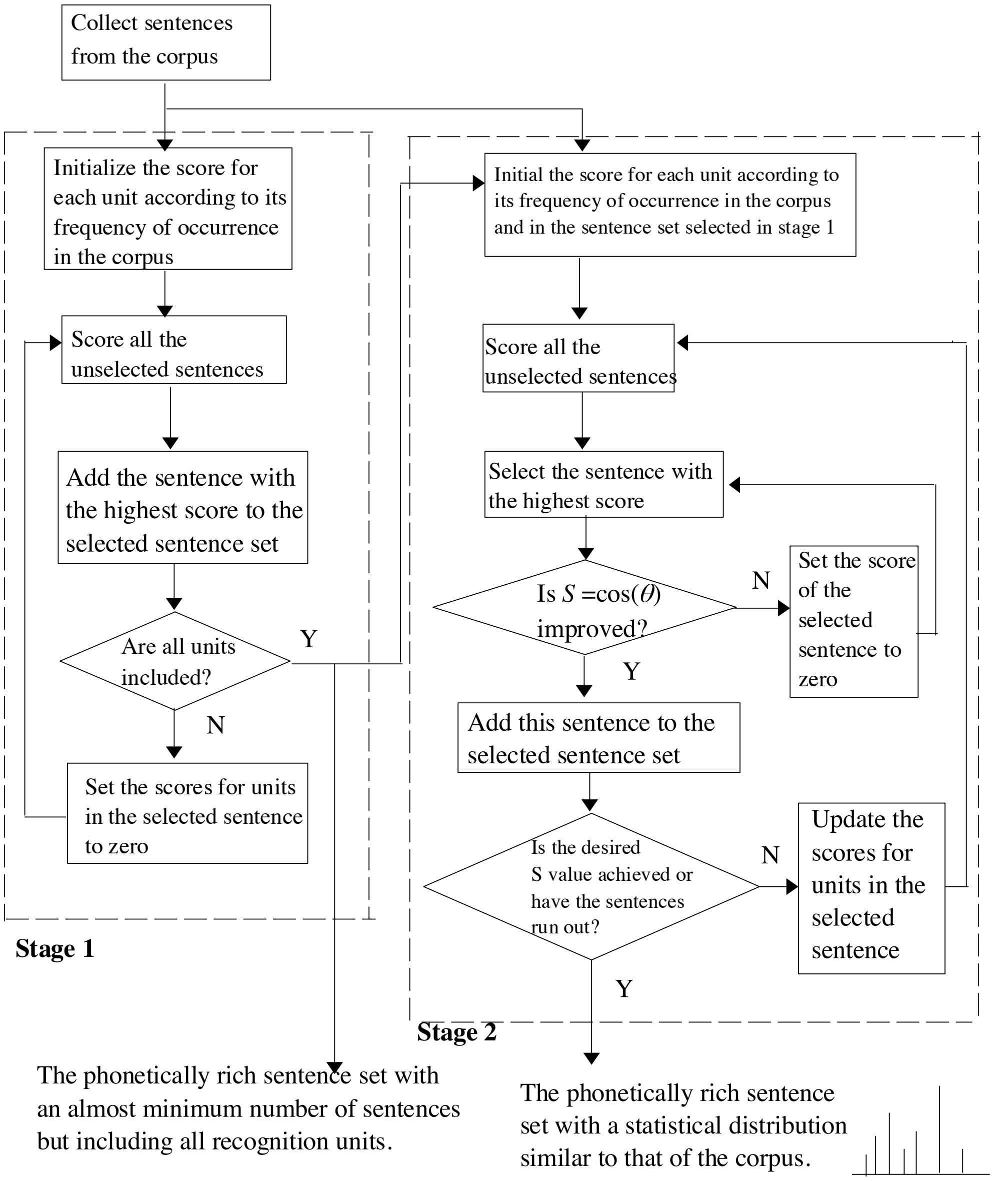

Collect sentences from the corpus
Initialize the score for each unit according to its frequency of occurrence in the corpus
Score all the unselected sentences
Add the sentence with the highest score to the selected sentence set
Are all units included?
Y
N
Set the scores for units in the selected sentence to zero
Stage 1
Initial the score for each unit according to its frequency of occurrence in the corpus and in the sentence set selected in stage 1
Score all the unselected sentences
Select the sentence with the highest score
Is S =cos(θ) improved?
N
Set the score of the selected sentence to zero
Y
Add this sentence to the selected sentence set
Is the desired S value achieved or have the sentences run out?
N
Update the scores for units in the selected sentence
Y
Stage 2
The phonetically rich sentence set with an almost minimum number of sentences but including all recognition units.
The phonetically rich sentence set with a statistical distribution similar to that of the corpus.

***Figure 3** The Flow Chart of the Two-stage Sentence Selection Algorithm*

The flow chart of our two-stage algorithm is shown in Figure 3, and the basic principles used in designing this algorithm can be described by the following rules:

(1) All recognition units used in the corpus should be included.

(2) Those sentences with a larger number of different recognition units should be

selected with higher priority.

(3) In the first stage, those sentences consisting of units with lower frequency of occurrence in the corpus should be selected with higher priority, so that the total number of sentences to cover all the units can be as small as possible.

(4) In the second stage, on the other hand, those sentences consisting of units with higher frequency of occurrence in the corpus should be selected with higher priority, so that the desired statistical distribution can be achieved as soon as possible.

In the first stage, the input is the whole text corpus, and the desired output is an almost smallest set of sentences, including all the necessary recognition units plus co-articulation effects. To achieve this goal, a score is first assigned to each unit (co-articulation effects should be included when defining context-dependent units), which is initialized as the reciprocal of its frequency of occurrence in the text corpus, so that rare units have higher priority for selection. A score is also defined for each sentence, which is calculated as the average of the scores of its component units but is modified using two weights. The first weight, defined as,

$$w_1 = \frac{\text{number of distinct units in a given sentence}}{\text{number of units in a given sentence}}, \qquad (1)$$

is higher for sentences with a larger number of distinct recognition units because such sentences should have higher scores and be selected with higher priority. The second weight is used to confine the selected sentences to the desired sentence length. Certainly, a long sentence can contain much richer contextual information than a short sentence. However, in general, it is difficult for people to utter long sentences with clear pronunciation. That is, the desired sentences should be neither too long nor too short. The second weight is, therefore, defined as,

$$w2 = \begin{cases} 1.0 & min\_length \le L \le max\_length \\ 0.5 & otherwise \end{cases} \qquad (2)$$

where L is the length (number of units) of a given sentence while *min_length* and *max_length* are the minimum and maximum constraints for the sentence length, respectively. In this paper, 6 for *min_length* and 12 for *max_length* are adopted. Once a sentence is selected, the scores of all the units contained in this sentence are immediately set to zero to avoid these units being selected again. The first stage of the algorithm thus recursively updates the scores of the units and of all the left unselected sentences and selects additional sentences with the highest score, until all the recognition units are included. In this way, an almost minimum number of sentences which includes all the recognition units can be obtained.

In the second stage of the algorithm, the input is the left unselected sentences in the text corpus and the set of sentences obtained in the first stage, and the desired output is a set of phonetically rich sentences with a statistical distribution for the units very similar to that of the original text corpus. In this stage, the score of each unit is re-defined in a different way. An additional down factor is first defined for each unit, which is proportional to the reciprocal of the number of times this unit appears in the original text corpus. This down factor is used to reduce the priority of a unit to be selected again after each selection. The initial score for each unit in the second stage is then defined as a constant subtracted by its down factor multiplied by the number of times it has been selected previously in the first stage. In this way, the units with higher frequency of occurrence in the original text corpus and lower frequency of occurrence in the set of sentences obtained in the first stage will have higher priority for selection. The rest of the algorithm is very similar to the first stage part. However, in this stage, a similarity measure S, as defined in equation (3), is used to estimate the degree to which the statistical distribution of the units in the selected phonetically rich sentence set is similar to that in the original text corpus:

$$S = \frac{\vec{V}_c \cdot \vec{V}_b}{\left|\vec{V}_c\right|\left|\vec{V}_b\right|} = \cos(\theta) \tag{3}$$

where $\vec{V}_c = \left[n_c(1), \ \ldots, \ n_c(i), \ \ldots n_c(N)\right]$,

$\vec{V}_b = [n_b(1), \ \ldots, \ n_b(i), \ \ldots n_b(N)]$, $n_c(i)$ is the number of times the i-th unit appears in the corpus, $n_b(i)$ is the number of times the i-th unit has been included in the currently selected phonetically rich sentence set, and N is the total number of different recognition units. Apparently, $\vec{V}_c$, $\vec{V}_b$ represent the statistical distribution of the units in the corpus and in the selected sentence set, respectively, S is the normalized inner product of $\vec{V}_c$ and $\vec{V}_b$, and θ is the angle between $\vec{V}_c$ and $\vec{V}_b$. When $S = 1$, i.e., $\vec{V}_c = k \,.\, \vec{V}_b$, the statistical distributions will be exactly identical. Now, the sentence with the highest score and on the same time can improve the similarity measure S is first added to the phonetically rich sentence set. Once a sentence is selected, the scores for all its component units are immediately subtracted by its down factor. By recursively selecting additional sentences one by one as described above until the desired similarity measure S is achieved, one can obtain a set of phonetically rich sentences with a statistical distribution similar to that of the text corpus to be used as a good training and evaluating set.

4.1 Detailed Description of the Two-stage Algorithm

Further details for each stage are given here. Except for N, $n_c(i)$, $n_b(i)$, and L, which have been defined above, all the symbols that will be used in the following are given first.

N_c:the total number of the recognition units in the corpus;

$s[i]$:the score for the i-th unit;

$d_s[i]$:the score down factor for the i-th unit.

4.1.1 The First Stage

The procedure in stage 1 is :

(1) Collect all the sentences from the corpus.

(2) Initialize the score for each unit:

$$s[i] = \frac{1}{n_c[i]}, \quad i = 1, \ \ldots, \ N. \tag{4}$$

(3) Score all the unselected sentences.

(4) Add the sentence with the highest score (*SENT*) to the selected sentence set.

(5) If all the units contained in the corpus are included, end stage 1 and go to stage 2

(6) Set the scores for units contained in *SENT* to zero;

$$s[i_k]=0, k=1, ...,L, i_k \text{ is the k-th unit of } SENT. \tag{5}$$

Then, go to step 3.

4.1.2 The Second Stage

The procedure in stage 2 is :

(1) Continue from stage 1.

(2) Initialize the score for each unit:

for i=1, . . . , N

$$s[i]=const$$

$$d_s[i]=\frac{s[i]}{n_c[i]}$$

$$s[i]=s[i]-d_s[i]\times n_b[i] \tag{6}$$

(3) Score all the unselected sentences.

(4) If the sentence with the highest score (*SENT*) can improve the similarity, add *SENT* to the selected sentence set. Otherwise, go to step 7.

(5) If the constraint for S is satisfied or if all the sentences in the corpus have run out, end stage 2.

(6) Update the scores for the units contained in *SENT*;

$$s[i_k]=s[i_k]-d_s[i_k],\ \ k=1,\ \ldots L,\ \ i_k \text{ is the k-th unit of SENT.} \tag{7}$$

Then, go to step 3.

(7) Set the score of the highest score sentence to zero and go to step 4.

5. Two Example Experiments for Extraction of Phonetically Rich Chinese Sentences

Two example experiments were performed to test the proposed algorithm. Both were designed to select a set of phonetically rich Chinese sentences to be used for continuous Mandarin speech recognition. Context-independent tonal syllables were chosen as the recognition units in the first experiment while context-dependent INITIALs and context-independent FINALs were chosen in the second experiment. Both examples were chosen simply due to their simplicity. The same algorithm can certainly be used if some other more complicated context-dependent units are needed, as long as the target units are defined. The Chinese text corpus used here consists of a total of 124,845 sentences (1,374,182 syllables), which is a subset of the corpus described in section 3.

In the first experiment, the recognition units chosen were the 1345 phonologically allowed context-independent tonal syllables (i.e., assuming inter-syllabic co-articulation is negligible) in Mandarin due to the monosyllabic structure of the Chinese language. The results are summarized in Table 12. It can be found that at the end of stage 1, only 366 sentences (2790 syllables) were sufficient to include all the recognition units (1345 tonal syllables), in which each tonal syllable appeared only about 2.5 times on average. These numbers correspond to very small percentages of the whole corpus (0.29% of sentences and 0.20% of syllables, respectively). Note that, if a larger text corpus is used, these percentages can be even smaller. A nice feature here is that the statistical distribution of the tonal syllables included in these 366 sentences obtained in stage 1 is already quite similar to that of the corpus ($S = \cos(\theta) = 0.9064$) because the tonal syllables with higher frequency of occurrence are very naturally carried over into the set selected in the first stage of the algorithm although in this stage, tonal syllables with lower frequency of occurrence have higher priority for selection. When the second stage was performed, on the other hand, the similarity measure $S = \cos(\theta)$ improved very quickly as more sentences were included. When 650 to 750 sentences were included, the statistical distribution was really very close to that of the corpus (S = 0.9931 and 0.9959, respectively) although still much less sentences (0.52% to 0.60%) were needed as compared to the whole corpus. Though the co-articulation effects were not considered in this example, it is obvious that the phonetically rich sentences with co-articulation effects can also be obtained if the context-dependent units are defined.

Stage	Selected sentences		Selected syllables		S	θ
	total number	% in corpus	total number	% in corpus	$\cos(\theta)$	(degrees)
1	366	0.293	2790	0.203	0.9064	24.990
2	400	0.320	3022	0.220	0.9410	19.777
	450	0.360	3377	0.246	0.9681	14.515
	500	0.400	3723	0.271	0.9802	11.433
	550	0.441	4067	0.296	0.9869	9.295
	600	0.481	4405	0.321	0.9907	7.808
	650	0.521	4744	0.345	0.9931	6.742
	700	0.561	5093	0.371	0.9949	5.817
	750	0.601	5477	0.399	0.9959	5.171

Table 12 *The Simulation Results for Selecting Phonetically Rich Chinese Sentences from a Corpus Using 1345 Phonologically Allowed Context-independent Tonal Syllables as the Recognition Units*

In the second experiment, the recognition units chosen were 113 context-dependent INITIALs and 41 context-independent FINALs due to the monosyllabic nature and the INITIAL/FINAL structure of Mandarin Chinese. As shown in Table 2(c), the 41 FINALs can be divided into 8 groups according to their beginning phonemes, and the FINALs in the same group can be assumed to have the same influence on their preceding INITIALs because they all have the same beginning phoneme. For example, the /s/ in the base syllables /sai/, /sau/, /san/, etc. is assumed to be the same in each case but different from the /s/ in the base syllables /su/, /suo/, etc. In this way, the 22 INITIALs can be expanded to 113 context-dependent INITIALs. In other words, the 113 context-dependent INITIALs are kind of "generalized diphones"; i.e., they depend on the group of following FINALs, but co-articulation with the FINALs of the previous syllables is assumed to be negligible. FINALs, on the other hand, are just assumed to be context-independent because the co-articulation effect on both sides of a FINAL is not significant. Therefore, this example only considered "intra-syllable" co-articulation with a "right-to-left" direction but not "inter-syllable" co-articulation. The results are listed in Table 13. It can be found that only 28 sentences (191 syllables) were sufficient to cover all the recognition units (113 context-dependent INITIALs and 41 context-independent FINALs) after the first stage was completed, in which each INITIAL appeared about 1.7 times and each FINAL about 4.7 times. Though the similarity measure $S=\cos(\theta)$ was not very high ($S=0.8068, \theta=36.211^{\circ}$) after the first stage was performed, the second stage could improve S even more quickly than in the previous example. When 80 to 100 sentences (515 to 639 syllables) were included, the statistical distribution was really very close to that of the corpus although still much less sentences (0.064% to 0.080%) were needed as compared to the whole corpus. Other phonetically rich sentences for more complicated context-dependent units (e.g., context-dependent INITIALs and FINALs considering both left and right contextual effects, or other context-dependent phone-like units) can certainly be selected using the same algorithm.

Stage	Selected sentences		Selected syllables		S	θ
	total number	% in corpus	total number	% in corpus	$\cos(\theta)$	(degrees)
1	28	0.022	191	0.0139	0.8068	36.211
2	30	0.024	203	0.0148	0.8469	32.120
	40	0.032	264	0.0192	0.9452	19.048
	50	0.040	327	0.0238	0.9735	13.232
	60	0.048	388	0.0282	0.9862	9.522
	70	0.056	450	0.0327	0.9919	7.279
	80	0.064	515	0.0375	0.9955	5.408
	90	0.072	577	0.0420	0.9971	4.383
	100	0.080	639	0.0465	0.9979	3.682

***Table 13** The Simulation Results for Selecting Phonetically Rich Chinese Sentences from a Corpus Using 113 Context-dependent INITIALs and 41 Context-independent FINALs as the Recognition Units.*

6. Conclusions

How to design a set of phonetically rich sentences to be used in efficiently training and evaluating a speech recognition system has become a very important issue in speech recognition research. In this paper, we have presented statistical analysis of various Mandarin acoustic units, such as syllables, tones, and INITIAL/FINALs, which have been widely adopted as the basic recognition units in many Mandarin speech recognition systems, based upon a very large Chinese text corpus collected from daily newspapers. Furthermore, we have proposed a two-stage algorithm to automatically extract phonetically rich sentences from a text corpus to be used in training and evaluating a speech recognition system. We have also proved the efficiency of this algorithm through two example experiments on selecting phonetically rich Chinese sentence sets from a Chinese text corpus. This algorithm can be applied to any language, any recognition task, and any pre-defined recognition units with co-articulation effects, as long as the text corpus defining the task is given.

Acknowledgments

The author would like to thank Mr. Yu-hsueh Chou and Mr. Yuan-cheng Chang for their contributions in programming. Special thanks are due to Prof. Lin-shan Lee for many helpful comments. Thanks are also due to the three anonymous reviewers for their valuable suggestions.

References

Akira Kurematsu, Kazuya Takeda, Yoshinori Sagisaka, Shigeru Katagiri, Hisao Kuwabara, and Kiyohiro Shikano, "ATR Japanese Speech Database As A Tool of Speech Recognition and Synthesis", *Speech Communication*, No. 9, 1990, pp. 365-374.

Bai B.-R., Tseng C.-Y., and Lee L.-S., "A Multi-phase Approach for Fast Spotting of Large Vocabulary Chinese Keywords from Mandarin Speech Using Prosodic Information", *ICASSP97*, Vol. 2, pp. 903-906.

Chang H.-Y., B. Chen Chou C.-S., and Liu C.-M., "Speaker-independent Mandarin Polysyllabic Word Recognition", *Int. Symp. on Signal Processing and Its Applications*, 1996.

Chen K.-J. and Liu S.-H., "Word Identification for Mandarin Chinese Sentences", *COLING92*, pp. 101-107.

CKIP group, "Analysis of Syntactic Categories for Chinese", *CKIP Technical Report, No. 93-05*, Institute of Information Science, Academia Sinica, Taipei, 1993.

Huang C.-C. and Wang J.-F., "A Mandarin Speech Dictation System Based on Neural Network and Language Processing Model", *IEEE Trans. on Consumer Electronics*, Vol. 40, No. 3, 1994, pp. 437-445.

Huang E.-F, Wang H.-C., and Soong F.-K., "A Fast Algorithm for Large Vocabulary Keyword Spotting Application", *IEEE Trans. on Speech and Audio Processing*, Vol. SAP-2, No. 3, 1994, pp. 449-452.

Lee K.-F., Hon H.-W., and R. Reddy, "An Overview of the SPHINX Speech Rcognition System", *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. 38, No. 1, Jan. 1990, pp. 35-45.

Lee L.-S. et al., "Golden Mandarin (I) - A Real-time Mandarin Speech Dictation Machine for Chinese Language with very Large Vocabulary", *IEEE Trans. on Speech and Audio Pro-*

cessing, Vol. 1, No. 2, April 1993, pp. 158-179.

Lee L.-S. et al., "Golden Mandarin (II) - An Improved Single-chip Real-time Mandarin Dictation Machine for Chinese Language with very Large Vocabulary", *ICASSP93*, pp. 503-506.

Lin B.-S., Wang H.-M., and Lee L.-S., "Key-phrase Understanding Framework Integrating Real World Knowledge with Speech Recognition with Initial Application in Voice Memo Systems for Mandarin Chinese", *IEEE TENCON97*, pp. 157-160.

Lyu Renyuan, Lee L.-S. et al., "Golden Mandarin (III) - A User-adaptive Prosodic-segment-based Mandarin Dictation Machine for Chinese Language with very Large Vocabulary", *CASSP95*, pp. 57-60.

Ney H., V. Steinbiss, R. Haeb-Umbach, B.-H. Tran, and U. Essen, "An Overview of the Philips Research System for Large-Vocabulary Continuous Speech Recognition", *Int. J. Pattern Recognition and Artificial Intelligence*, Vol. 8, No. 1, Feb. 1994, pp. 33-70.

Shen J.-L., *Improved Mandarin Dictation: New Technologies and Golden Mandarin (III) Windows 95 Version*, Ph. D. dissertation, National Taiwan University, Dec. 1996.

Tseng C.-Y., "A Phonetically Oriented Speech Database for Mandarin Chinese", *proc. International Congress of Phonetic Sciences*, Vol. 3, 1995, pp. 326-329.

Wang H.-C., "MAT - a project to collect Mandarin speech data through telephone networks in Taiwan", *Int. J. of Computational Linguistic & Chinese Language Processing*, Vol. 2, No. 1, February 1997, pp. 73-90.

Wang H.-M. and Lee L.-S., "Tone Recognition for Continuous Mandarin Speech with Limited Training Data Using Selected Context-dependent Hidden Markov Models", *J. of The Chinese Institute of Engineers*, Vol. 17, No. 6, 1994, pp. 775-784.

Wang H.-M., Lee L.-S. et al., "Complete Recognition of Continuous Mandarin Speech for Chinese Language with very Large Vocabulary but Limited Training Data", *IEEE Trans. on Speech and Audio Processing*, Vol. 5, No. 2, March 1997, pp. 195-200.

Wang Y.-R. and Chen S.-H., "Tone Recognition of Continuous Mandarin Speech Assited with Prosodic Information", *J. Acoustic Society of America*, Vol. 96, No. 5, 1994, pp. 2637-2645.

Yu S.-M. and Liu C.-S., "The Construction of Phonetically Balanced Chinese Sentences", *Telecommunication Laboratories Technical Journal*, R.O.C., Vol. 28, No. 1, Jan. 1990, pp. 84-91.

Zue Victor, Stephanie Seneff, and James Glass, "Speech Database Development at MIT: TIMIT and Beyond", *Speech Communication*, No. 9, 1990, pp. 365-374.