

NYU Language Modeling Experiments for the 1996 CSR Evaluation

Satoshi Sekine, Andrew Borthwick and Ralph Grishman

Computer Science Department
New York University
715 Broadway, 7th floor
New York, NY 10003, USA

ABSTRACT

This paper describes NYU's effort toward improving recognition accuracy for the 1996 ARPA Large Vocabulary Continuous Speech Recognition evaluation. We are trying to develop different kinds of language models including longer-range models and a linguistically motivated model. For the system described here, we used as a starting point the scores produced by SRI's acoustic and language models. These are linearly combined with the scores produced by the NYU language models. This paper also describes some experiments we tried which were not used in the official experiment, including experiments with perplexity minimization, Maximum Entropy modeling and parsing.

1. Introduction

This paper describes NYU's effort toward improving recognition accuracy for the 1996 ARPA Large Vocabulary Continuous Speech Recognition evaluation. Our goal has been to study some longer-range language models and determine whether they can be a useful component of the language models used for speech recognition. We will explain the model we used for the official evaluation, done in collaboration with SRI. (SRI's system is described in [1]) The technique used for the official evaluation is essentially the same as last year's model. We used a topic coherence model, cache model and weighted cache model.

This paper also describes some experiments we tried which were not used in the official experiment, including experiments with perplexity minimization, Maximum Entropy modeling and parsing.

2. Topic coherence model - official evaluation -

We worked jointly with SRI this year. For the system described here for the official evaluation, we used as a starting point the scores produced by SRI's acoustic and language models. These are linearly combined with the scores produced by the NYU language models, and then the hypothesis with the highest total score is selected

2.1. Sublanguage model

Our approach can be briefly summarized as follows. The topic or subject matter of an article influences its linguistic properties, such as word choice and co-occurrence patterns; in effect it gives rise to a very specialized "sublanguage" for that topic. We try to find the sublanguage to which the article belongs based on the sentences already recognized. At a stage in transcription-mode speech recognition processing, some words in the other utterances are selected as keywords. Then, based on these keywords, similar articles are retrieved from a large corpus by a method similar to information retrieval. The re-

trieved articles are assembled into a sublanguage "mini-corpus" for the current article. We then analyze the mini-corpus in order to determine word preferences which will be used in analyzing the sentence currently being processed. The details of each step were described in last year's paper [6], although some minor parameters were set to different values to fit this year's evaluation.

2.2. Weighted cache model

We combined this sublanguage model with a cache model and a weighted cache model. Our "traditional" cache model [which is the same as the model we used last year] assigns a score to every word based on the log of the ratio between the unigram frequency of the word in the current document and the unigram frequency of the word in the corpus as a whole:

$$F'(w) = \text{Number of occurrences of } w \text{ in current article} \quad (1)$$

$$N' = \sum_{w \in \text{article}} F'(w) \quad (2)$$

$$F(w) = \text{Number of occurrences of } w \text{ in corpus} \quad (3)$$

$$M = \sum_{w \in \text{corpus}} F(w) \quad (4)$$

$$CScore(w) = \log \left(\frac{F'(w)/N'}{F(w)/M} \right) \quad (5)$$

The purpose of the denominator in this formula is to compensate for the fact that high-frequency words are more likely to appear in the cache than low-frequency words. For instance, the occurrence of a high-frequency word like "Clinton" supplies much less new information to the language model than does the appearance of a relatively low-frequency word such as "Schwarzenegger".

Although our experiments have shown that $CScore(w)$ is useful, it doesn't make use of the fact that some words tend to be highly concentrated in a few articles whereas other words are likely to be spread fairly evenly over the corpus. Consider the following two medium-frequency words:

Word	$F(w)$	$DF_1(w)$	$DF_{2+}(w)$	$E_1(w)$	$E_{2+}(w)$
second	51707	24273	9775	0.5187	0.8065
japan	50066	8103	8076	2.0945	3.1960

Definitions:

$$DF_j(w) = \text{Number of docs containing } w \text{ } j \text{ times}$$

$$DF_{j+}(w) = \text{Number of docs containing } w \text{ } j \text{ times or more}$$

$$E_1(w) = \frac{F(w)}{DF_{1+}(w)} - 1$$

$$E_{2+}(w) = \frac{F(w) - DF_1(w)}{DF_{2+}(w)} - 2$$

The two quantities which we used for our weighted cache prediction were $E_1(w)$ and $E_{2+}(w)$ —the expected number of reoccurrences of w given one appearance of w in the document and given 2+ appearances. These quantities are similar to those described in [10]. Note that we are using the same formula for 2+ appearances of w as for 2 appearances. We did this due to an intuition that further appearances probably contained no new information and also for simplicity of implementation.

It can be easily seen from the table that these quantities seem to model real phenomena because their values for “japan” are much higher than for “second” even though the two have about the same unigram frequency. This matches our intuition that “japan” is much more likely to be the topic of an article than is “second”.

We combined these quantities together using the following formula:

$$E(w) = \begin{cases} F'(w) = 0 & : 0 \\ F'(w) = 1 & : E_1(w) \\ F'(w) \geq 2 & : E_{2+}(w) \end{cases}$$

$$DocTot = \sum_{w:w \in \text{article}} E(w)$$

$$WCScore(w) = \log \left(\frac{E(w)/DocTot}{F(w)/M} \right)$$

In combining these three knowledge sources (sublanguage, cache, and weighted cache), we found from experiments on minimizing the error rate of the devtest data that we achieved our best results by using all three sources (see Figure 1).

	Word Errors	Improvement
SRI (baseline)	7273	
SRI + SL	7180	-93
SRI + cache	7173	-100
SRI + SL, cache	7126	-147
SRI + SL, w-cache	7136	-137
SRI + SL, cache, w-cache	7114	-159
SRI + SL, cache via M.E.	7224	-49

Table 1: Devtest Results

2.3. Result

The absolute improvement using the sublanguage component over SRI’s system is 0.3%, from 33.3% to 33.0%, as shown in Table 1. The absolute improvement is small; however, there is a limit to the improvement we can obtain, because the N-best sentences don’t always contain the correct candidate. It is important to see the difference between the number of errors produced by the base system and the minimum number of errors obtainable by choosing the N-best hypothesis with minimum error for each sentence. (We will call the latter error rate “MNE” for “minimal N-best errors”.) Although we don’t have the precise number for MNE for the 1996 evaluation, based on our estimate from dev data, we can suggest that our achievement is about 5% of the MNE (possible improvement). We believe that the result is satisfactory, because there are a lot of word errors

System	SRI	SRI+NYU
F0	26.4%	26.0%
F1	33.0%	32.5%
F2	31.7%	32.6%
F3	34.7%	34.2%
F4	38.5%	38.4%
F5	34.4%	31.1%
FX	48.3%	48.1%
F0	33.3%	33.0%

Figure 1: Formal Result

unrelated to the article topic, for example function word replacement (“a” replaced by “the”), or deletion or insertion of topic unrelated words (missing “over”).

2.4. Comparison to prior results

We have been working on topic coherence models for three years. The improvement we made each year is relatively small, e.g. 0.3% to 0.6% in absolute word error rate. However, it’s important to observe that we did obtain consistent improvements with this technique; this increases our confidence in the significance of our result.

Furthermore, the improvements were achieved with different corpora (WSJ, NAB, and BN) and different speech systems (BBN, SRI), as shown in Figure 2. This is also encouraging, because it demonstrates that the sublanguage technique indeed can work in such different environments.

Test (Partner)	baseline	NYU result	Absolute Improve.	Relative Improve.
96 BN (SRI)	33.3	33.0	0.3	(5%)
95 NAB-P0 (SRI)	24.6	24.0	0.6	10.4%
95 NAB-C0 (SRI)	9.7	9.4	0.3	5.6%
94 WSJ (BBN)	11.0	10.6	0.4	-

Figure 2: Results (History)

2.5. Related work

There have been several related efforts in the ARPA speech community. Table 2 shows some of the recent work which uses topic coherent techniques, including cache model and topic clustering methods. Because the evaluations were made on different test sets and conditions, a direct comparison is not possible. We have to be very careful about the conclusions we draw from the table. In general, we can find improvements using these techniques, although many are relatively small (except for the CMU experiment (94), which uses only long texts and different conditions). We summarize the techniques in three categories:

- Cache
Use previously uttered word information to supplement the language model. The weighted cache model also uses information about the differing likelihoods of words to reoccur in an article.

Site (Year)	Description	Result	Ref.
IBM (91)	cache model		[2]
CMU (94)	trigger model	19.9– >17.8	[3]
BU (93-94)	clustering (4 topic LM)	11.3– >11.2	[4]
NYU (94-96)	sublanguage, cache and weighted cache model	11.0– >10.6	[5]
		24.6– >24.0 33.3– >33.0	[6]
CMU (96)	hand clustering (5883 topic)	0.1,0.6% improve. in 2 story	[7]
SRI (96)	clustering (4 topic LM)	33.1– >33.0	[8]
CU (96)	cache model	27.7– >27.5	[9]

Table 2: Related works

In transcription mode, the information in the following input can also be used.

- **Dynamic Topic Adaptation (trigger, sublanguage)**
Dynamically consult a database to build a language model for the topic. The data can be structured in advance (trigger model) or the raw text data can be retrieved and analyzed on demand (sublanguage model), but in either case the set of topics is not defined in advance.
- **Clustering Language Model**
Prepare language models for several topics which are defined in advance (automatically or by hand). Then find the topic of the current segment and use the language model of the topic (or possibly a mix of several language models, also combined with the general language model).

3. Perplexity Minimization

We combine our components among themselves and relative to the SRI acoustic and n-gram components by using a simple linear combination of the log of the scores or probabilities produced by each component. These relative weights are determined by minimizing the error rate of devtest data. We were concerned, though, that the devtest data might be too small for this sort of training and that the problem would be exacerbated as we added additional linguistic components.

To get around this problem, we reformulated our cache and sublanguage models to produce probabilities rather than scores and ran some preliminary perplexity minimization experiments on a single day of WSJ data, which represented a 73,000 word corpus vs the 8,000 words which we had in the '95 devtest data. These experiments showed that the interpolation of cache and sublanguage probabilities with the baseline trigram probabilities caused a big decrease in perplexity, but we got no improvement in error rate when the weights which minimized perplexity were used on the devtest data.

Surprised by this result, we reran our perplexity experiments on the devtest text data and got perplexity and word error figures for over 200 different relative weightings of trigram, cache, and sublanguage values. A representative slice of this three-dimensional grid can be seen in Figure 3, which shows perplexity and word error rates for various weightings of the sublanguage/cache component relative to a standard backoff trigram component [16]. Note that in the chart the sublanguage/cache ratio is fixed at 4:6 and that the point “0.00”

represents a purely trigram model. As the figure shows, a big de-

weight for SL/cache	perplexity	Word Errors
0.00	151.9	720
0.03	131.2	716
0.06	128.4	715
0.09	127.8	719
0.12	128.2	721
0.15	129.3	724
0.18	130.9	727

Table 3: Perplexity and Error Rates

crease in perplexity might correspond to a minimal decrease in error rate (the 0.06 weighting) or an increase in errors (the 0.12 weighting). Furthermore, the perplexity minimum was fairly flat across a broad range, offering little guidance on the optimal relative values. We concluded from this experiment that perplexity is not necessarily a good guide to minimizing word error rate.

The reader may have noticed that the results achieved by the probabilistic approach were significantly worse than those produced by the original “scoring-based” model. This is probably due to various features which were left out of the experimental probability-based model. For instance, the scoring-based model used the entire remainder of the document as context for determining a sentence’s cache and sublanguage scores whereas the probabilistic model just used the preceding sentences in the document. It also seemed possible that the scoring-based formulae might be working better than the probabilistic formulae. Another possible explanation is that seeking to optimize the error rate by searching for a linear combination of log probabilities (or scores) may be better than doing the same with a combination of “unlogged” probabilities, as we did in the probabilistic model. We mention these differences just to point out that the probabilistic model cannot be directly compared with the scoring model. Since our results indicated that our probabilistic approach was not helpful, we ended up using the scoring-based model in the evaluation.

4. Maximum Entropy Experiments

Maximum entropy modeling (M.E.) offers some of the same benefits of the perplexity minimization method in that it allows us to train on large text corpora rather than on the smaller amount of n-best data for which we have acoustic data available. More importantly, though, M.E. gives us a new way of constructing these models and of combining them in a non-linear fashion.

Consider, for instance, the cache scoring formula of equation 5. This formula was developed according to our intuition of the nature of cache word repetition, but it is vulnerable to criticism on other intuitive grounds. For instance, does it make sense that the (unlogged) score for a word should double when a word has been seen twice as many times in an article? Furthermore, our team has had continuing internal debates about how to handle the interaction of the cache and sublanguage models: i.e. should the sublanguage model predict cache words or should it leave the prediction of those words entirely to the cache component?

M.E. theory [13] [14] offers an intuitively and theoretically satisfying answer to these sorts of questions which vex language modelers. When using M.E. for language modeling, one identifies a set of

linguistically significant “constraints” and a training corpus and then the M.E. algorithm builds a model which is guaranteed to:

- Conform to all of the constraints (assuming they are consistent with each other)
- Have the maximum entropy (i.e. be the “flattest”) of all models which conform to these constraints
- Maximize the probability of the training corpus, subject to the constraints (this is only true under certain conditions, which we adhered to in these experiments)

For our experiments, we used a set of constraints which closely mirrored the phenomena we were trying to capture in our previous cache and sublanguage modeling experiments:

$$\begin{aligned} Cache_k &= P(w|w \text{ seen } k \text{ times in article}) \\ SL_k^{df} &= P(w|w \text{ occurred in } k \text{ of the 50 similar articles} \\ &\quad \text{retrieved by the sublanguage module}) \\ SL_k^{wc} &= P(w|w \text{ has } k \text{ occurrences in the 50 similar ar-} \\ &\quad \text{ticles}) \end{aligned}$$

The M.E. algorithm will build a model in which the conditional probabilities of these features will conform, on average, with those found in the training corpus. This can be expressed more precisely in the following way, using the feature family $Cache_k$ as an example. First define an indicator function which is a function of the document history h and the current word w :

$$C_k(h, w) = \begin{cases} 1 & \text{if } k \text{ instances of } w \text{ have appeared in the} \\ & \text{current article prior to the current in-} \\ & \text{stance of } w \\ 0 & \text{otherwise} \end{cases}$$

Now we observe in the training corpus that

$$\sum_{(h,w)} \tilde{P}(h, w) C_k(h, w) = \alpha_k$$

We then constrain our M.E. language model to only consider conditional models, $P(w|h)$, which conform to this constraint:

$$\sum_{(h,w)} \tilde{P}(h) P(w|h) C_k(h, w) = \alpha_k$$

One departure of this work from that of other work in the field [14] is that we build a very small, and hence computationally tractable model, using only c. 200 constraints/parameters. Rosenfeld, by contrast, built a model which had c. 2.2 million parameters. The primary reason for the difference is that we are leaving n -gram constraints out of the model, whereas Rosenfeld incorporated them into his. We think that we may be paying a penalty in performance by doing this, but we hope that we will nevertheless squeeze significant benefits out of the model while avoiding the very heavy computational requirements which Rosenfeld reported—roughly two weeks machine time on 15 DEC/Alpha workstations.

Our preliminary results with these experiments were that we achieved only 31% of the gain which we achieved by using the conventional methods (see Figure 1). While these results might seem to be discouraging, we believe that they are due, in part, to the fact that we have not yet had sufficient time to experiment with the techniques.

Since we implemented this using a publicly available M.E. toolkit [15] which permits basically any knowledge source to be used as input so long as it can be parameterized along the lines shown above, we think that we may have the ability to integrate a large number of different linguistic sources into a single, unified model. Among the sources which we are thinking of integrating are:

- An M.E. formulation of our weighted cache component
- Some formulation of unigram, bigram, and trigram features which avoids a massive explosion in the number of parameters
- A parsing score as derived from the Apple Pie Parser (see next section)
- Miscellaneous linguistic features which would allow us to experiment with the effect of integrating a large number of diverse knowledge sources.

5. Parsing

As part of our effort to apply natural language techniques in order to improve recognition accuracy, we are developing a corpus-based statistical parser [11]. It is a probabilistic, bottom-up, best-first search chart parser, and its grammar is acquired from syntactically bracketed corpus. The special feature of the parser is that the number of non-terminals is relatively small (5 in the current version) in order to capture larger context. This parser is publically available [12].

Recently, we implemented a technique to incorporate the probabilities of lexical dependencies into the parser. We created a simple set of rules to identify the head of each constituent, and assigned dependency relationships between the head and all the other elements. This relationship is actually a long distance, syntactically motivated bigram (for example, between a verb and the head of its subject). In some cases, this dependency bigram can work better than the usual bigram, because the relationship is syntactically meaningful, and not just between consecutive words. However, the only currently available large syntactically tagged corpus is the University of Pennsylvania Tree Bank; we used the Wall Street Journal portion of the Tree Bank to acquire the lexical dependency probabilities. One of the serious and unavoidable problems is the limited size of the training corpus. Compared to the corpus size typically used for bigram training, the training size for the dependency relationships is significantly smaller. One idea for tackling this problem in the future is to use the parser in order to create a relatively reliable tagged corpus. We have found that the approach using the dependency relationships produces good performance for analyzing written text. The typical accuracy measurement (recall and precision of bracketing) improves about 2% compared to the parsing result without dependency relationships.

Because the domain of the training corpus is business newspaper articles, we decided that we would initially try the parsing scheme on the 1995 speech evaluation data from North American Business News domain rather than the 1996 (Broadcast News domain) evaluation.

5.1. Binary Comparison

First, in order to assess the ability of the parsing technique in speech recognition, we ran a ‘binary comparison’ experiment. From the N-best sentences, the best candidate based on SRI’s acoustic and language model scores (which we will call ‘SRI-best’), and the correct sentence (‘correct’) are extracted. Both of the sentences for each utterance are parsed and the scores are compared. The difference of the

parsing score is compared with the difference of the trigram score (Table 4). In the table, only those sentences where the correct sentence is in SRI's N-best and the correct sentence is not SRI's best sentence are reported. Our hope is that the parser will consistently prefer the correct sentence over SRI's best, and indeed we observed that in 60% of the cases the correct sentence had the better parsing score. Furthermore, we note that this subset of the eval data represents sentences on which the traditional (trigram) language model did not do so well; it preferred the correct sentence in fewer than 40% of the cases.

	trigram favors correct sent.	trigram favors SRI-best
Parser favors correct	16	23
Parser favors SRI-best	9	17

Table 4: Comparison between parser and trigram

In addition, we examined some of the individual sentence pairs (some of which are listed in the Appendix). In the remainder of this section, we will indicate the category of the result by using the position in the table (i.e. top-right or bottom-left) In the bottom-left category — examples which are not good for the parsing model — we found some bugs in the grammar, as well as some inevitable cases. where local evidence is as important as, or more important than, wide syntactic context. For example, in the third pair of sentences in Appendix, the parser prefers the parent company shareholders rather than the parent company's shareholders. This is because the part-of-speech sequence DT NN NN NNS is more likely than DT NN NN POS NNS (here, DT=determiner, NN=singular noun, NNS=plural noun and POS=possessive). However, if you look at the words, the correct sentence is at least as plausible as the other hypothesis (as the trigram model predicted). We can find several instances of this kind in the bottom-left category.

By looking at the 23 instances in the top-right category — where the parser predicted correctly while the trigram model did not — we find a number of encouraging examples. Six example are listed in the Appendix. For example, in the first sentence, macdonnell . . . , SRI's best candidate, has no verb, yet the trigram score for the candidate is better than for the correct sentence. In the second sentence they say . . . , there are too many verbs in SRI's best candidate. This is exactly what we expected to achieve with a parser. In other words, sometimes wide context is more important for picking the correct words than local (trigram) context .

The other categories (16 top-left and 17 bottom-right in the table) are harmless; adding parsing score to trigram score in these cases does not affect the ranking of the two sentences. Many such cases are to be expected because syntactic context often includes local evidence.

Outside of this table, we found an interesting example. It concerns out-of-vocabulary words (in particular, proper nouns) and an example is shown in the Appendix under "other" category. It contains an OOV sequence of long proper nouns ("noriyuki matsushima"), but as these nouns are not in the vocabulary, the speech system produced an unusual sequence of words ("nora you keep matsui shima"). We could not calculate a trigram score for the correct hypothesis, but as you can imagine the parser assigned a much better score to the cor-

rect sentence. So, it may be interesting for future work to use the technique of parsing in order to try to identify these mistakes on out-of-vocabulary words.

5.2. Evaluation

Although we found some promising evidence in the binary comparison experiment, we found no improvement in speech evaluation when the parsing scores were linearly combined with the other sentence scores. This is understandable, because now we have 19 competitors (we used 20-best) rather than a single competitor in the binary experiment; there could be some other hypothesis which is syntactically more plausible but includes more word errors.

6. Conclusion

We have found consistent improvements in speech recognition accuracy based on a topic-coherence model. In particular, the improvements under different test conditions increase our confidence in the significance of our overall result.

We found some suggestive evidence that the parser may be able to help, although it is not yet at the point of improving recognition accuracy. As it seems promising, it is worth pushing this line of research. This will include improving the parser and also adapting the parser to the recognition task. In particular, because the output style of the speech recognizer is not the same as the written text, we should make some adjustments to the grammar and dictionary. For example, the recognizer output does not have commas or quotation marks, which are significant clues in written text parsing, so the grammar needs to be adjusted accordingly.

7. Acknowledgments

The work reported here was supported by the Defense Advanced Research Projects Agency under contract DABT63-93-C-0058 from the Department of the Army. We would like to thank our collaboration partners at SRI, in particular Andreas Stolcke and Ananth Sankar. We would also like to thank Erik Sven Ristad of Princeton University for his rapid responses to our queries about his Maximum Entropy Modeling Toolkit. Finally, we would like to thank Slava Katz, who worked on this project as a consultant.

References

1. A. Sankar, A. Stolcke, L. Heck and F.Weng "SRI H4-PE System Overview" in *this proceedings* (1997)
2. F Jelinek, B Meriardo, S Roukos, and M Strauss: "A Dynamic Language Model for Speech Recognition" *Proceedings of DARPA Speech and Natural Language Workshop* (1991)
3. Ronald Rosenfeld "Adaptive Statistical Language Modeling" *Proceedings of Human Language Technology Workshop* (1994)
4. M.Ostendorf, F.Richardson, R.Iyer, A.Kannan, O.Ronen and R.Bates "The 1994 BU NAB News Benchmark System" *Proceedings of the ARPA Spoken Language Systems Technology Workshop* (1995)
5. Satoshi Sekine, John Sterling and Ralph Grishman "NYU/BBN 1994 CSR evaluation" *Proceedings of the ARPA Spoken Language Systems Technology Workshop* (1995)
6. Satoshi Sekine and Ralph Grishman "NYU Language Modeling Experiments for the 1995 CSR Evaluation" *Proceedings of the DARPA Speech Recognition Workshop* (1996)

7. Kristie Seymore, Stanley Chen, Maxine Eskenazi and Roni Rosenfeld "Language and Pronunciation Modeling in the CMU 1996 Hub 4 Evaluation" *Proceedings of the DARPA Speech Recognition Workshop* (1997)
8. Fuliang Weng, Andreas Stolcke and Ananth Sankar "Hub-4 Language Modeling using Domain Interpolation and Data Clustering" *Proceedings of the DARPA Speech Recognition Workshop* (1997)
9. Steve Young, Mark Gales, David Pye and Phil Woodland "HTK Broadcast News Language Model" *Proceedings of the DARPA Speech Recognition Workshop* (1997)
10. Slava M. Katz "Distribution of content words and phrases in text and language modeling" *Natural Language Engineering, Vol.2 Part.1, pp15-60* (1996)
11. Satoshi Sekine, Ralph Grishman "A Corpus-based Probabilistic Grammar with Only Two Non-terminals" *Proceedings of the Fourth International Workshop on Parsing Technologies* (1995)
12. Satoshi Sekine "Apple Pie Parser: home page" <http://cs.nyu.edu/cs/projects/proteus/app> (1996)
13. Edwin T. Jaynes "Information Theory and Statistical Mechanics" *Physics Reviews 106, pp620-630*, (1957)
14. Ronald Rosenfeld "Adaptive Statistical Language Modeling: A Maximum Entropy Approach" *CMU Technical Report CMU-CS-94-138* (1994)
15. Eric Sven Ristad "Maximum Entropy Modeling Toolkit, release 1.5 Beta" <ftp://ftp.cs.princeton.edu/pub/packages/mem/> (1997)
16. Slava M. Katz "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer" *IEEE Transactions on Acoustics, Speech, and Signal Processing* (1987)

Appendix: Binary Comparison Examples

Parser and trigram scores are shown in parentheses. Smaller numbers are better.

C: correct sentence S: SRI-best candidate

Parser and Trigram both favor SRI-best

C: some dealers of foreign cars also lowered
their japanese prices (448,655)
S: some dealers of foreign cars also lowered
the japanese prices (424,614)

C: the problem isn't gridlock he says the
wheels are out of alignment (598,646)
S: the problem is in gridlock he says the
wheels are out of alignment (567,625)

Trigram favor Correct, but Parser favor SRI-best (Bad example)

C: board would review distributing the
remaining shares in the gold subsidiary to
the parent company's shareholders (1328,1306)
S: board would review distributing the
remaining shares in the gold subsidiary to
the parent company shareholders (1254,1333)

Trigram favor SRI-best, but Parser favor Correct (Good example)

C: mcdonnell douglas corporation has built
helicopter parts ... (1360,1548)
S: mcdonnell douglas corporation and bell
helicopter parts ... (1404,1491)

C: they are interested in commodities as
a new asset class van says (521,731)
S: they are interested in commodities says
a new asset class van says (560,720)

C: weary of worrying about withdrawal
charges if you want to leave ... (1132,1273)
S: weary of worrying about withdraw all
charges if you want to leave ... (1210,1202)

C: this scenario as they say on t.v. is
based on a true story (550,649)
S: this scenario as a say on t.v. is
based on a true story (576,644)

C: indirect foreign ownership is limited to 25%
(613,723)
S: in direct foreign ownership is limited to 25%
(695,709)

C: even some lawyers now refer clients to
mediators offering to review the mediated
agreement and provide advice if needed
(1045,1255)

S: even some lawyers now refer clients to
mediators offering to review the mediated
agreement can provide advice if needed
(1067,1253)

Others

C: the may figures show signs of improving sales
said noriyuki matsushima (951,?)
S: the may figures show signs of improving sales
said nora you keep matsui shima (1211,?)