Analyzing the Reading Comprehension Task

Amit Bagga
GE Corporate Research and Development
1 Research Circle
Niskayuna, NY 12309
bagga@crd.ge.com

Abstract

In this paper we describe a method for analyzing the reading comprehension task. First, we describe a method of classifying facts (information) into categories or levels; where each level signifies a different degree of difficulty of extracting a fact from a piece of text containing it. We then proceed to show how one can use this model the analyze the complexity of the reading comprehension task. Finally, we analyze five different reading comprehension tasks and present results from this analysis.

1 Introduction

Recently there has been a spate of activity for building question-answering systems (QA systems) driven largely by the recently organized QA track at the Eighth Text Retrieval Conference (TREC-8) (Harman, 1999). This increase in research activity has also fueled research in a related area: building Reading Comprehension systems (Hirschman and others, 1999). But while a number of successful systems have been developed for each of these tasks, little, if any, work has been done on analyzing the complexities of the tasks themselves. In this paper we describe a method of classifying facts (information) into categories or levels; where each level signifies a different degree of difficulty of extracting a fact from a piece of text containing it. We then proceed to show how one can use this model the analyze the complexity of the reading comprehension task. Finally, we analyze five different reading comprehension tasks and present results from this analysis.

2 The Complexity of Extracting a Fact From Text

Any text document is a collection of facts (information). These facts may be explicitly or implicitly stated in the text. In addition, there are "easy" facts which may be found in a single sentence (example: the name of a city) as well as "difficult" facts which are spread across several sentences (example: the reason for a particular event).

For a computer system to be able to process text documents in applications like information extraction (IE), question answering, and reading comprehension, it has to have the ability to extract facts from text. Obviously, the performance of the system will depend upon the type of fact it has to extract: explicit or implicit, easy or difficult, etc. (by no means is this list complete). In addition, the performance of such systems varies greatly depending on various additional factors including known vocabulary, sentence length, the amount of training, quality of parsing, etc. Despite the great variations in the performances of such systems, it has been hypothesized that there are facts that are simply harder to extract than others (Hirschman, 1992).

In this section we describe a method for estimating the complexity of extracting a fact from text. The proposed model was initially used to analyze the information extraction task (Bagga and Biermann, 1997). In addition to verifying Hirschman's hypothesis, the model also provided us with a framework for analyzing and understanding the performance of several IE systems (Bagga and Biermann, 1998). We have also proposed using this model to analyze the complexity of the QA task which is related to both the IE, and the reading comprehension tasks (Bagga et al., 1999). The remainder of this section describes the model in detail, and provides a sample application of the model to an IE task. In the following section, we discuss how this model can be used to analyze the reading comprehension task.

2.1 Definitions

Network:
A *network* consists of a collection of nodes interconnected by an accompanying set of arcs. Each node denotes an object and each arc represents a binary relation between the objects. (Hendrix, 1979)

A Partial Network:
A *partial network* is a collection of nodes interconnected by an accompanying set of arcs where the collection of nodes is a subset of a collection of nodes forming a network, and the accompanying set of arcs is a subset of the set of arcs accompanying the set of nodes which form the network.

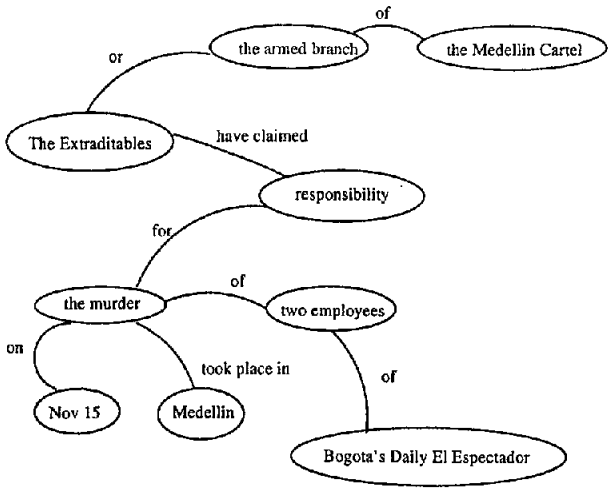

Figure 1: A Sample Network

Figure 1 shows a sample network for the following piece of text:

> "The Extraditables," or the Armed Branch of the Medellin Cartel have claimed responsibility for the murder of two employees of Bogota's daily El Espectador on Nov 15. The murders took place in Medellin.

2.2 The Level of A Fact

The level of a fact, F, in a piece of text is defined by the following algorithm:

1. Build a network, S, for the piece of text.
2. Identify the nodes that are relevant to the fact, F. Suppose $\{x_1, x_2, \ldots, x_n\}$ are the nodes relevant to F. Let s be the partial network consisting of the set of nodes $\{x_1, x_2, \ldots, x_n\}$ interconnected by the set of arcs $\{t_1, t_2, \ldots, t_k\}$.
   We define the *level* of the fact, F, *with respect to* the network, S to be equal to k, the number of arcs linking the nodes which comprise the fact F in s.

2.2.1 Observations

Given the definition of the level of a fact, the following observations can be made:

- The level of a fact is related to the concept of "semantic vicinity" defined by Schubert et. al. (Schubert and others, 1979). The *semantic vicinity* of a node in a network consists of the nodes and the arcs reachable from that node by traversing a small number of arcs. The fundamental assumption used here is that "the knowledge required to perform an intellectual task generally lies in the semantic vicinity of the concepts involved in the task" (Schubert and others, 1979).

  The level of a fact is equal to the number of arcs that one needs to traverse to reach all the concepts (nodes) which comprise the fact of interest.
- A level-0 fact consists of a single node (i.e. no transitions) in a network.
- A level-k fact is a *union* of k level-1 facts.
- Conjunctions/disjunctions increase the level of a fact.
- A higher level fact is likely to be harder to extract than a lower level fact.
- A fact appearing at one level in a piece of text may appear at some other level in the same piece of text.
- The level of a fact in a piece of text depends on the granularity of the network constructed for that piece of text. Therefore, the level of a fact with respect to a network built at the word level (i.e. words represent objects and the relationships between the objects) will be greater than the level of a fact with respect to a network built at the phrase level (i.e. noun groups represent objects while verb groups and preposition groups represent the relationships between the objects).

2.2.2 Examples

Let S be the network shown in Figure 1. S has been built at the phrase level.

- The city mentioned, in S, is an example of a level-0 fact because the "city" fact consists only of one node "Medellin."
- The type of attack, in S, is an example of a level-1 fact.

We define the *type of attack* in the network to be an attack designator such as "murder," "bombing," or "assassination" with one modifier giving the victim, perpetrator, date, location, or other information.

In this case the type of attack fact is composed of the "the murder" and the "two employees" nodes and their connector. This makes the type of attack a level-1 fact.

The type of attack could appear as a level-0 fact as in "the Medellin bombing" (assuming that the network is built at the phrase level) because in this case both the attack designator (bombing) and the modifier (Medellin) occur in the same node. The type of attack fact occurs as a level-2 fact in the following sentence (once again assuming that the network is built at the phrase level): "10 people were killed in the offensive which included several bombings." In this case there is no direct connector between the attack designator (several bombings) and its modifier (10 people). They are connected by the intermediatory "the offensive" node; thereby making the type of attack a level-2 fact. The type of attack can also appear at higher levels.

- In S, the date of the murder of the two employees is an example of a level-2 fact.
  This is because the attack designator (the murder) along with its modifier (two employees) account for one level and the arc to "Nov 15" accounts for the second level.

  The date of the attack, in this case, is not a level-1 fact (because of the two nodes "the murder" and "Nov 15") because the phrase "the murder on Nov 15" does not tell one that an attack actually took place. The article could have been talking about a seminar on murders that took place on Nov 15 and not about the murder of two employees which took place then.
- In S, the location of the murder of the two employees is an example of a level-2 fact.
  The exact same argument as the date of the murder of the two employees applies here.
- The complete information, in S, about the victims is an example of a level-2 fact because to know that two employees of Bogota's Daily El Espectador were victims, one has to know that they were murdered. The attack designator (the murder) with its modifier (two employees) accounts for one level, while the connector between "two employees" and "Bogota's Daily El Espectador" accounts for the other.

2.3 Building the Networks

As mentioned earlier, the level of a fact for a piece of text depends on the network constructed for the text. Since there is no unique network corresponding to a piece of text, care has to be taken so that the networks are built consistently.

We used the following algorithm to build the networks:

1. Every article was broken up into a non-overlapping sequence of noun groups (NGs), verb groups (VGs), and preposition groups (PGs). The rules employed to identify the NGs, VGs, and PGs were almost the same as the ones employed by SRI's FASTUS system[1].
2. The nodes of the network consisted of the NGs while the transitions between the nodes consisted of the VGs and the PGs.
3. Identification of coreferent nodes and prepositional phrase attachments were done manually.

The networks are built based largely upon the syntactic structure of the text contained in the articles. However, there is some semantics encoded into the networks because identification of coreferent nodes and preposition phrase attachments are done manually.

Obviously, if one were to employ a different algorithm for building the networks, one would get different numbers for the level of a fact. But, if the algorithm were employed consistently across all the facts of interest and across all articles in a domain, the numbers on the level of a fact would be consistently different and one would still be able to analyze the relative complexity of extracting that fact from a piece of text in the domain.

3 Example: Analyzing the Complexity of an Information Extraction Task

In order to validate our model of complexity we applied it to the Information Extraction (IE) task, or the Message Understanding task (DAR, 1991), (DAR, 1992), (ARP, 1993), (DAR, 1995), (DAR, 1998). The goal of an IE task is to extract prespecified facts from text and fill in predefined templates containing labeled slots.

We analyzed the complexity of the task used for the Fourth Message Understanding Conference (MUC-4) (DAR, 1992). In this task, the participants were asked to extract the following facts from articles describing terrorist activities in Latin America:

- The type of attack.
- The date of the attack.
- The location of the attack.

[1]We wish to thank Jerry Hobbs of SRI for providing us with the rules of their partial parser.

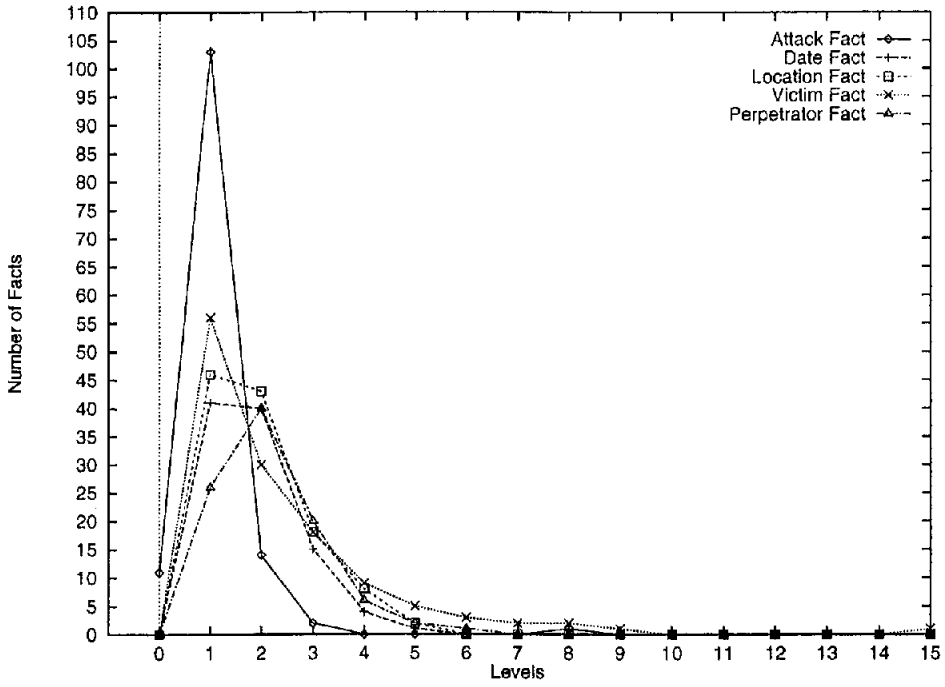

Figure 2: MUC-4: Level Distribution of Each of the Five Facts

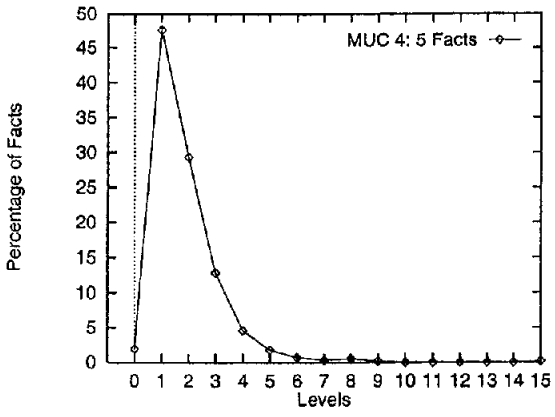

Figure 3: MUC-4: Level Distribution of the Five Facts Combined

- The victim (including damage to property).
- The perpetrator(s) (including suspects).

We analyzed a set of 100 articles from the MUC-4 domain each of which reported one or more terrorist attacks. Figure 2 shows the level distribution for each of the five facts. A closer analysis of the figure shows that the "type of attack" fact is the easiest to extract while the "perpetrator" fact is the hardest (the curve peaks at level-2 for this fact). In addition, Figure 3 shows the level distribution of the five facts combined. This figure gives some indication of the complexity of the MUC-4 task because it shows that almost 50% of the MUC-4 facts occur at level-1. The expected level of the five facts in the MUC-4 domain was 1.74 (this is simply the weighted average of the level distributions of the facts). We define this number to be the *Task Complexity* for the MUC-4 task. Therefore, the MUC-4 task can now be compared to, say, the MUC-5 task by comparing their *Task Complexities*. In fact, we computed the Task Complexity of the MUC-5 task and discovered that it was equal to 2.5. In comparison, an analysis, using more "superficial" features, done by Beth Sundheim, shows that the nature of the MUC-5 EJV task is approximately twice as hard as the nature of the MUC-4 task (Sundheim, 1993). The features used in the study included vocabulary size, the average number of words per sentence, and the average number of sentences per article. More details about this analysis can be found in (Bagga and Biermann, 1998).

4 Analyzing the Reading Comprehension Task

The reading comprehension task differs from the QA task in the following way: while the goal of the QA task is to find answers for a set of questions from *a collection of documents*, the goal of the reading comprehension task is to find answers to a set of questions from *a single related document*. Since the QA task involves extracting answers from a collection of documents, the complexity of this task depends on the expected level of occurrence of the answers of the questions. While it is theoretically possible to compute the average level of *any* fact in the entire

Test	# of sentences	avg # of levels/sent	avg # of corefs/sent	# of questions	avg # of levels/answer	avg # of corefs/answer
Basic	9	4.11	2.33	8	3.75	2.25
Basic-Interm	13	2.69	2.39	6	3.33	2.50
Intermediate	56	3.50	2.55	9	4.44	3.33
Interm-Adv	17	6.47	1.00	6	7.83	1.33
Advanced	27	6.93	2.08	10	8.20	2.90

Figure 4: Summary of Results

document collection, it is not humanly possible to analyze every document in such large collections to compute this. For example, the TREC collection used for the QA track is approximately 5GB. However, since the reading comprehension task involves extracting the answers from a single document, it is possible to analyze the document itself in addition to computing the level of the occurrence of each answer. Therefore, the results presented in this paper will provide both these values.

4.1 Analysis and Results

We analyzed a set of five reading comprehension tests offered by the English Language Center at the University of Victoria in Canada [2]. These five tests are listed in increasing order of difficulty and are classified by the Center as: Basic, Basic-Intermediate, Intermediate, Intermediate-Advanced, and Advanced. For each of these tests, we calculated the level number of each sentence in the text, and the level number of the sentences containing the answers to each question for every test. In addition, we also calculated the number of coreferences present in each sentence in the texts, and the corresponding number in the sentences containing each answer. It should be noted that we were forced to calculate the level number of the sentences containing the answer as opposed to calculating the level number of the answer itself because several questions had only true/false answers. Since there was no way to compute the level numbers of true/false answers, we decided to calculate the level numbers of the sentences containing the answers in order to be consistent. For true/false answers this implied analyzing all the sentences which help determine the truth value of the question.

Figure 4 shows for each text, the number of sentences in the text, the average level number of a sentence, the average number of coreferences per sentence, the number of questions corresponding to the test, the average level number of each answer, and the average number of coreferences per answer.

The results shown in Figure 4 are consistent with the model. The figure shows that as the difficulty level of the tests increase, so do the corresponding level numbers per sentence, and the answers. One conclusion that we can draw from the numbers is that the Basic-Intermediate test, based upon the analysis, is slightly more easy than the Basic test. We will address this issue in the next section.

The numbers of coreferences, surprisingly, do no increase with the difficulty of the tests. However, a closer look at the types of coreference shows that while most of the coreferences in the first two tests (Basic, and Basic-Intermediate) are simple pronominal coreferences (he, she, it, etc.), the coreferences used in the last two tests (Intermediate-Advanced, and Advanced) require more knowledge to process. Some examples include *marijuana* coreferent with *the drug*, *hemp* with *the pant*, etc. Not being able to capture the complexity of the coreferences is one, among several, shortcomings of this model.

4.2 A Comparison with Qanda

MITRE [3] ran its Qanda reading comprehension system on the five tests analyzed in the previous section. However, instead of producing a single answer for each question, Qanda produces a list of answers listed in decreasing order of confidence. The rest of this section describes an evaluation of Qanda's performance on the five tests and a comparison with the analysis done in the previous section.

In order to evaluate Qanda's performance on the five tests we decided to use the Mean Reciprocal Answer Rank (MRAR) technique which was used for evaluating question-answering systems at TREC-8 (Singhal, 1999). For each answer, this techniques assigns a score between 0 and 1 depending on its rank in the list of answers output. The score for answer, i, is computed as:

$$\text{Score}_i = \frac{1}{\text{rank of answer}_i}$$

If no correct answer is found in the list, a score of 0 is assigned. Therefore, MRAR for a reading comprehension test is the sum of the scores for answers corresponding to each question for that test.

Figure 5 summarizes Qanda's results for the five tests. The figure shows, for each test, the number of questions, the cumulative MRAR for all answers for the test, and the average MRAR per answer.

[2] http://web2.uvcs.uvic.ca/elc/studyzone/index.htm

[3] We would like to thank Marc Light and Eric Breck for their help with running Qanda on our data.

Test	# of questions	MRAR for all answers	avg MRAR per answer
Basic	8	2.933	0.367
Basic-Interm	6	3.360	0.560
Intermediate	9	2.029	0.226
Interm-Adv	6	1.008	0.168
Advanced	10	7.833	0.783

Figure 5: Summary of Qanda's Results

The results from Qanda are more or less consistent with the analysis done earlier. Except for the Advanced test, the average Mean Reciprocal Answer Rank is consistent with the average number of levels per sentence (from Figure 4). It should be pointed out that the system performed significantly better on the Basic-Intermediate Test compared to the Basic test consistent with the numbers in Figure 4. However, contrary to expectation, Qanda performed exceedingly well on the Advanced test answering 7 out of the 10 questions with answers whose rank is 1 (i.e. the first answer among the list of possible answers for each question is the correct one). We are currently consulting the developers of the system for conducting an analysis of the performance on this test in more detail.

5 Shortcomings

This measure is just the beginning of a search for useful complexity measures. Although the measure is a big step up from the measures used earlier, it has a number of shortcomings. The main shortcoming is the ambiguity regarding the selection of nodes from the network regarding the fact of interest. Consider the following sentence: "This is a report from the Straits of Taiwan. .... Yesterday, China test fired a missile." Suppose we are interested in the location of the launch of the missile. The ambiguity here arises from the fact that the article does not explicitly mention that the missile was launched in the Straits of Taiwan. The decision to infer that fact from the information present depends upon the person building the network.

In addition, the measure does not account for the following factors (the list is not complete):

coreference: If the extraction of a fact requires the resolution of several coreferences, it is clearly more difficult than an extraction which does not. In addition, the degree of difficulty of resolving coreferences itself varies from simple exact matches, and pronominal coreferences, to ones that require external world knowledge.

frequency of answers: The frequency of occurrence of facts in a collection of documents has an impact on the performance of systems.

occurrence of multiple (similar) facts: Clearly, if several similar facts are present in the same article, the systems will find it harder to extract the correct fact.

vocabulary size: Unknown words present some problems to systems making it harder for them to perform well.

On the other hand, no measure can take into account all possible features in natural language. Consider the following example. In an article, suppose one initially encounters a series of statements that obliquely imply that the following statement is false. Then the statement is given: "Bill Clinton visited Taiwan last week." Processing such discourse requires an ability to perfectly understand the initial series of statements before the truth value of the last statement can be properly evaluated. Such complete understanding is beyond the state of the art and is likely to remain so for many years.

Despite these shortcomings, the current measure does quantify complexity on one very important dimension, namely the number of clauses (or phrases) required to specify a fact. For the short term it appears to be the best available vehicle for understanding the complexity of extracting a fact.

6 Conclusions

In this paper we have described a model that can be used to analyze the complexity of a reading comprehension task. The model has been used to analyze five different reading comprehension tests, and the paper presents the results from the analysis.

References

ARPA. 1993. *Fifth Message Understanding Conference (MUC-5)*, San Mateo, August. Morgan Kaufmann Publishers, Inc.

Amit Bagga and Alan W. Biermann. 1997. Analyzing the Complexity of a Domain With Respect To An Information Extraction Task. In *Tenth International Conference on Research on Computational Linguistics (ROCLING X)*, pages 175–194, August.

Amit Bagga and Alan W. Biermann. 1998. Analyzing the Performance of Message Understanding Systems. *Journal of Computational Linguis-*

tics and Chinese Language Processing, 3(1):1–26, February.

Amit Bagga, Wlodek Zadrozny, and James Pustejovsky. 1999. Semantics and Complexity of Question Answering Systems: Towards a Moore's Law for Natural Language Engineering. In *1999 AAAI Fall Symposium Series on Question Answering Systems*, pages 1–10, November.

DARPA. 1991. *Third Message Understanding Conference (MUC-3)*, San Mateo, May. Morgan Kaufmann Publishers, Inc.

DARPA. 1992. *Fourth Message Understanding Conference (MUC-4)*, San Mateo, June. Morgan Kaufmann Publishers, Inc.

DARPA: TIPSTER Text Program. 1995. *Sixth Message Understanding Conference (MUC-6)*, San Mateo, November. Morgan Kaufmann Publishers, Inc.

DARPA: TIPSTER Text Program. 1998. *Seventh Message Understanding Conference (MUC-7)*. http://www.muc.saic.com/proceedings/muc_7_toc.html, April.

D. K. Harman, editor. 1999. *Eighth Text REtrieval Conference (TREC-8)*. National Institute of Standards and Technology (NIST), U.S. Department of Commerce, National Technical Information Service, November.

Gary G. Hendrix. 1979. Encoding Knowledge in Partitioned Networks. In Nicholas V. Findler, editor, *Associative Networks*, pages 51–92. Academic Press, New York.

Lynette Hirschman et al. 1999. Deep Read: A Reading Comprehension System. In *37th Annual Meeting of the Association of Computational Linguistics*, pages 325–332, June.

Lynette Hirschman. 1992. An Adjunct Test for Discourse Processing in MUC-4. In *Fourth Message Understanding Conference (MUC-4)* (DAR, 1992), pages 67–77.

Lenhart K. Schubert et al. 1979. The Structure and Organization of a Semantic Net for Comprehension and Inference. In Nicholas V. Findler, editor, *Associative Networks*, pages 121–175. Academic Press, New York.

Amit Singhal. 1999. Question Answering Track at TREC-8. *http://www.research.att.com/~singhal/qa-track-spec.txt*, November.

Beth M. Sundheim. 1993. Tipster/MUC-5 Information Extraction System Evaluation. In *Fifth Message Understanding Conference (MUC-5)* (ARP, 1993), pages 27–44.