A Semi-Automatic Evaluation Scheme: Automated Nuggetization for Manual Annotation

Liang Zhou, Namhee Kwon, and Eduard Hovy
Information Sciences Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292
{liangz, nkwon, hovy}@isi.edu

Abstract

In this paper we describe automatic information nuggetization and its application to text comparison. More specifically, we take a close look at how machine-generated nuggets can be used to create evaluation material. A semi-automatic annotation scheme is designed to produce gold-standard data with exceptionally high inter-human agreement.

1 Introduction

In many natural language processing (NLP) tasks, we are faced with the problem of determining the appropriate granularity level for information units. Most commonly, we use sentences to model individual pieces of information. However, more NLP applications require us to define text units smaller than sentences, essentially decomposing sentences into a collection of phrases. Each phrase carries an independent piece of information that can be used as a standalone unit. These finer-grained information units are usually referred to as *nuggets*.

When performing within-sentence comparison for redundancy and/or relevancy judgments, without a precise and consistent breakdown of nuggets we can only rely on rudimentary *n*-gram segmentations of sentences to form nuggets and perform subsequent *n*-gram-wise text comparison. This is not satisfactory for a variety of reasons. For example, one *n*-gram window may contain several separate pieces of information, while another of the same length may not contain even one complete piece of information.

Previous work shows that humans can create nuggets in a relatively straightforward fashion. In the PYRAMID scheme for manual evaluation of summaries (Nenkova and Passonneau, 2004), machine-generated summaries were compared with human-written ones at the nugget level. However, automatic creation of the nuggets is not trivial. Hamly et al. (2005) explore the enumeration and combination of all words in a sentence to create the set of all possible nuggets. Their automation process still requires nuggets to be manually created *a priori* for reference summaries before any summary comparison takes place. This human involvement allows a much smaller subset of phrase segments, resulting from word enumeration, to be matched in summary comparisons. Without the human-created nuggets, text comparison falls back to its dependency on n-grams. Similarly, in question-answering (QA) evaluations, gold-standard answers use manually created nuggets and compare them against system-produced answers broken down into n-gram pieces, as shown in POURPRE (Lin and Demner-Fushman, 2005) and NUGGETEER (Marton and Radul, 2006).

A serious problem in manual nugget creation is the inconsistency in human decisions (Lin and Hovy, 2003). The same nugget will not be marked consistently with the same words when sentences containing multiple instances of it are presented to human annotators. And if the annotation is performed over an extended period of time, the consistency is even lower. In recent exercises of the PYRAMID evaluation, inconsistent nuggets are flagged by a tracking program and returned back to the annotators, and resolved manually.

Given these issues, we address two questions in this paper: First, how do we define nuggets so that they are consistent in definition? Secondly, how do

Proceedings of NAACL HLT 2007, Companion Volume, pages 217–220,
Rochester, NY, April 2007.

we utilize automatically extracted nuggets for various evaluation purposes?

2 Nugget Definition

Based on our manual analysis and computational modeling of nuggets, we define them as follows:
Definition:

- A nugget is predicated on either an *event* or an *entity*.
- Each nugget consists of two parts: the anchor and the content.

The anchor is either:

- the head noun of the entity, or
- the head verb of the event, plus the head noun of its associated entity (if more than one entity is attached to the verb, then its subject).

The content is a coherent single piece of information associated with the anchor. Each anchor may have several separate contents.

When a nugget contains nested sentences, this definition is applied recursively. Figure 1 shows an example. Anchors are marked with square brackets. If the anchor is a verb, then its entity attachment is marked with curly brackets. If the sentence in question is a compound and/or complex sentence, then this definition is applied recursively to allow decomposition. For example, in Figure 1, without recursive decomposition, only two nuggets are formed: 1) "[girl] working at the bookstore in Hollywood", and 2) "{girl} [talked] to the diplomat living in Britain". In this example, recursive decomposition produces nuggets with labels 1-a, 1-b, 2-a, and 2-b.

2.1 Nugget Extraction

We use syntactic parse trees produced by the Collins parser (Collins, 1999) to obtain the structural representation of sentences. Nuggets are extracted by identifying subtrees that are descriptions for entities and events. For entity nuggets, we examine subtrees headed by "NP"; for event nuggets, subtrees headed by "VP" are examined and their corresponding subjects (siblings headed by "NP") are treated as entity attachments for the verb phrases.

Sentence:
The girl working at the bookstore in Hollywood talked to the diplomat living in Britain.

Nuggets are:

1) [girl] working at the bookstore in Hollywood
    a. [girl] working at the bookstore
    b. [bookstore] in Hollywood
2) {girl} [talked] to the diplomat living in Britain
    a. {girl} [talked] to the diplomat
    b. [diplomat] living in Britian

Anchors:

1) [girl]
    a. [girl]
    b. [bookstore]
2) {girl} [talked]: talked is the anchor verb and girl is its entity attachment.
    a. {girl} [talked]
    b. [diplomat]

Figure 1. Nugget definition examples.

3 Utilizing Nuggets in Evaluations

In recent QA and summarization evaluation exercises, manually created nuggets play a determinate role in judging system qualities. Although the two task evaluations are similar, the text comparison task in summarization evaluation is more complex because systems are required to produce long responses and thus it is hard to yield high agreement if manual annotations are performed. The following experiments are conducted in the realm of summarization evaluation.

3.1 Manually Created Nuggets

During the recent two Document Understanding Confereces (DUC-05 and DUC-06) (NIST, 2002–2007), the PYRAMID framework (Nenkova and Passonneau, 2004) was used for manual summary evaluations. In this framework, human annotators select and highlight portions of reference summaries to form a pyramid of summary content units (SCUs) for each docset. A pyramid is constructed from SCUs and their corresponding popularity scores—the number of reference summaries they appeared in individually. SCUs carrying the same information do not necessarily have the same surface-level words. Annotators need to make the decisions based on semantic equivalence among

various SCUs. To evaluate a peer summary from a particular docset, annotators highlight portions of text in the peer summary that convey the same information as those SCUs in previously constructed pyramids.

3.2 Automatically Created Nuggets

We envisage the nuggetization process being automated and nugget comparison and aggregation being performed by humans. It is crucial to involve humans in the evaluation process because recognizing semantically equivalent units is not a trivial task computationally. In addition, since nuggets are system-produced and can be imperfect, annotators are allowed to reject and re-create them. We perform record-keeping in the background on which nugget or nugget groups are edited so that further improvements can be made for nuggetization.

The evaluation scheme is designed as follows:

For *reference summaries* (per docset):

- Nuggets are created for all sentences;
- Annotators will group equivalent nuggets.
- Popularity scores are automatically assigned to nugget groups.

For *peer summaries*:

- Nuggets are created for all sentences;
- Annotators will match/align peer's nuggets with reference nugget groups.
- Recall scores are to be computed.

3.3 Consistency in Human Involvement

The process of creating nuggets has been automated and we can assume a certain level of consistency based on the usage of the syntactic parser. However, a more important issue emerges. When given the same set of nuggets, would human annotators agree on nugget group selections and their corresponding contributing nuggets? What levels of agreement and disagreement should be expected? Two annotators, one familiar with the notion of nuggetization (C1) and one not (C2), participated in the following experiments.

Figure 2 shows the annotation procedure for reference summaries. After two rounds of individual annotations and consolidations and one final round of conflict resolution, a set of gold-standard

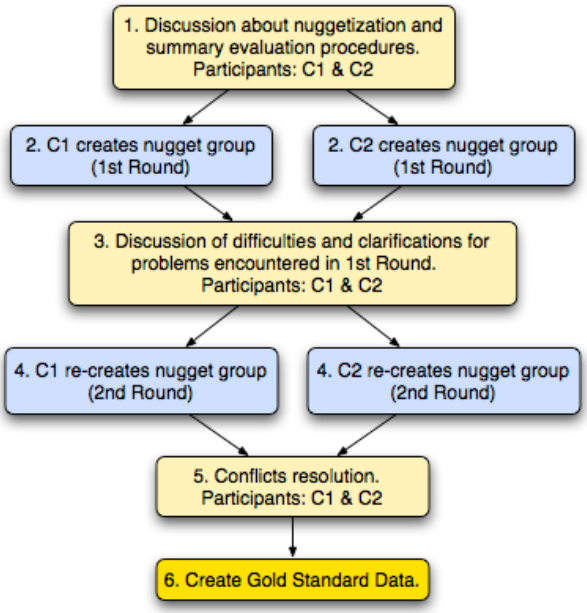

Figure 2. Reference annotation and gold-standard data creation.

nugget groups is created for each docset and will be subsequently used in peer summary annotations. The first round of annotation is needed since one of the annotators, C2, is not familiar with nuggetization. After the initial introduction of the task, concerns and questions arisen can be addressed. Then the annotators proceed to the second round of annotation. Naturally, some differences and conflicts remain. Annotators must resolve these problems during the final round of conflict resolution and create the agreed-upon gold-standard data. Previous manual nugget annotation has used one annotator as the primary nugget creator and another annotator as an inspector (Nenkova and Passonneau, 2004). In our annotation experiment, we encourage both annotators to play equally active roles. Conflicts between annotators resulting from ideology, comprehension, and interpretation differences helped us to understand that complete agreement between annotators is not realistic and not achievable, unless one annotator is dominant over the other. We should expect a 5-10% annotation variation.

In Figure 3, we show annotation comparisons from first to second round. The *x*-axis shows the nugget groups that C1 and C2 have agreed on. The *y*-axis shows the popularity score a particular nugget group received. Selecting from three reference summaries, a score of three for a nugget group indicates it was created from nuggets in all three

summaries. The first round initially appears successful because the two annotators had 100% agreement on nugget groups and their corresponding scores. However, C2, the novice nuggetizer, was much more conservative than C1, because only 10 nugget groups were created. The geometric mean of agreement on all nugget group assignment is merely 0.4786. During the second round, differences in group-score allocations emerge, 0.9192, because C2 is creating more nugget groups. The geometric mean of agreement on all nugget group assignment has been improved to 0.7465.

After the final round of conflict resolution, gold-standard data was created. Since all conflicts must be resolved, annotators have to either convince or be convinced by the other. How much change is there between an annotator's second-round annotation and the gold-standard? Geometric mean of agreement on all nugget group assignment for C1 is 0.7543 and for C2 is 0.8099. Agreement on nugget group score allocation for C1 is 0.9681 and for C2 is 0.9333. From these figures, we see that while C2 contributed more to the gold-standard's nugget group creations, C1 had more accuracy in finding the correct number of nugget occurrences in reference summaries. This confirms that both annotators played an active role. Using the gold-standard nugget groups, the annotators performed 4 peer summary annotations. The agreement among peer summary annotations is quite high, at approximately 0.95. Among the four, annotations on one peer summary from the two annotators are completely identical.

4 Conclusion

In this paper we have given a concrete definition for information nuggets and provided a systematic implementation of them. Our main goal is to use these machine-generated nuggets in a semi-automatic evaluation environment for various NLP applications. We took a close look at how this can be accomplished for summary evaluation, using nuggets created from reference summaries to grade peer summaries. Inter-annotator agreements are measured to insure the quality of the gold-standard data created. And the agreements are very high by following a meticulous procedure. We are currently preparing to deploy our design into full-scale evaluation exercises.

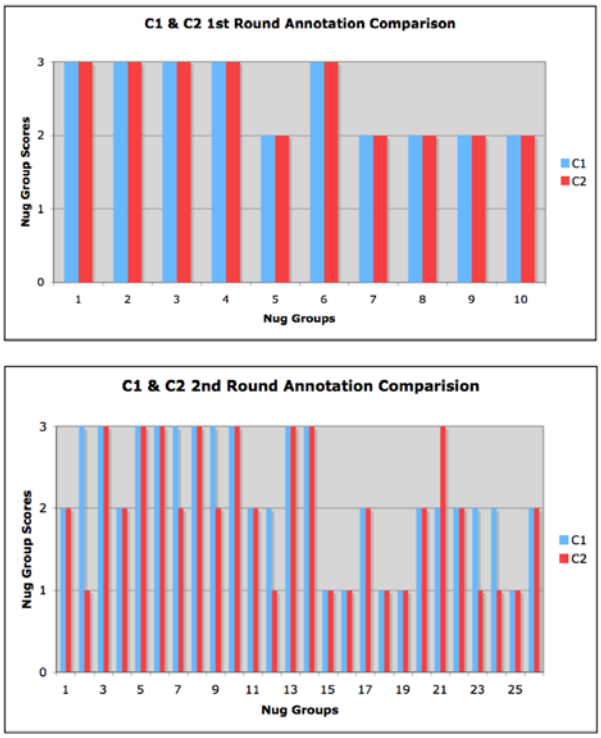

Figure 3. Annotation comparisons from 1st to 2nd round.

References

Collins, M. 1999. Head-driven statistical models for natural language processing. *PhD Dissertation*, University of Pennsylvania.

Hamly, A., A. Nenkova, R. Passonneau, and O. Rambow. 2005. Automation of summary evaluation by the pyramid method. In *Proceedings of* RANLP.

Lin, C.Y. and E. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of* NAACL-HLT.

Lin, J. and D. Demner-Fushman. 2005. Automatically evaluating answers to definition questions. In *Proceedings of* HLT-EMNLP.

Marton, G. and A. Radul. 2006. Nuggeteer: automatic nugget-based evaluation using description and judgments. In *Proceedings* NAACL-HLT.

Nenkova, A. and R. Passonneau. 2004. Evaluating content selection in summarization: the pyramid method. In *Proceedings* NAACL-HLT.

NIST. 2001–2007. Document Understanding Conference. www-nlpir.nist.gov/projects/duc/index.html.