Chapter 2

State of the Art

The aim of this chapter is to present previous research in AS. Due to the interdisciplinary nature of the Summarization field, several disciplines are involved, such as Linguistics, Logic, Philosophy, Psychology, Anthropology and Information Science, among others. In this chapter we focus in the Artificial Intelligence (AI) point of view, specially when using automatic NLP techniques.

The human production of abstracts has been studied for a long time. (Borko and Bernier 1975), a well known book for library science students and professional abstractors, states: "a well-written abstract must convey both the significant content and the character of the original work". This is an interesting definition but hard to implement using automated procedures and even harder to validate.

When the objective is to automate the summarization process, the study of human summarizers is a line of research that provides useful insight to establish a model to produce abstracts automatically. Some work has been carried out in that direction, to try to identify the phases of the human abstracting process, finding tasks and rules:

- (Cremmins 1982) states that the abstracting process consists of mental acts performed in four approximate separated stages: focusing on the basic features of the materials to be abstracted, identifying relevant information; extracting; organizing and reducing the relevant information into a coherent unit, usually one paragraph long; and refining the completed abstract through editing.
- Following the model presented by (van Dijk and Kintsch 1978), (Brown and Day 1983) propose a methodology for abstracting based on five rules: removing irrelevant information; overcoming redundancy; producing concept superarrangement; selecting the thematic sentence representing the text, if possible; and constructing the abstract.
- (Endres-Niggemeyer 1989) differentiates two kinds of rules for producing abstracts: ana-

lytical ones (reducing and selecting), with which the salient facts are obtained and condensed, and synthesis rules (clarifying, reorganizing and stylizing), with which the text of the abstract is produced. (Pinto Molina 1995) also agrees in these two phases for abstract production, proposing an integrated model based on three operative stages: reading-understanding, selection and interpretation, and synthesis. The synthesis phase is the most delicate, the instructions are stated at a very high conceptual level and are difficult to be implemented in an automatic procedure. Using Pinto's words, "It is practically impossible to establish techniques of synthesis that are valid for all types of documents and abstractor".

In the same direction as (Pinto Molina 1995), many researchers in automated creation of summaries of one or more texts have decomposed the process in three distinct stages (Sparck-Jones 1999; Mani and Maybury 1999; Hahn and Mani 2000; Hovy 2001; Mani 2001):

- *Analyzing* the input text to obtain text representation.
- *Transforming* it into a summary representation.
- *Synthesizing* an appropriate output form to generate the summary text.

(Sparck-Jones 2001) divides the Text Summarization (TS) problem into:

- *Locating* the relevant fragments, i.e. sentences, paragraphs, passages or utterances of a source text.
- *Ranking* these fragments by relevance.
- *Producing* a summary, either an *extract*, if the summary is composed by literal fragments of text, or an *abstract*, if it is generated.

(Hovy 2005) identifies the following stages:

- *Topic identification*: Recognition of the most important unit(s) (words, sentences, paragraphs, etc) containing a particular subject wrote about it or discussed in the text. He states that "Most of the systems today embody the first stage only".
- *Interpretation*: Fusion of concepts. He states that "No system can perform interpretation without prior knowledge about the domain."
- *Summary generation*: Summary content selection by abstracting and/or information extraction, with the exception that "Extract summaries do not require generation stage."

Instead of trying to automate all kinds of human summarization tasks, this thesis is oriented to automate summarization tasks that are really expensive or difficult for humans.

Research on AS has a long tradition, specially in Single Document Summarization (SDS), since the summaries based on relative word frequencies proposed by (Luhn 1958) or the disagreement between humans when perceiving relevance detected by (Rath et al. 1961). During a first period (50s – 70s), AS was applied mainly to scientific text using statistical techniques, and the first commercial applications were developed (Pollock and Zamora 1975). During the 80s, symbolic AI techniques were used to summarize short texts, narrative and news trying to model strategies used by humans. In the 90s some hybrid systems appeared mainly for news and scientific text. In recent years, new tasks have been considered, such as headline generation or Multi-Document Summarization (MDS) for different domain or scenario: news, law, medicine, e-mail, web-pages, using hand-held devices and summarizing documents in different languages or from different media. Many comparative studies can befound in the literature, among the most comprehensive, (Paice 1990); (Zechner 1997); (Hovy and Marcu 1998); (Sparck-Jones 1999); (Tucker 1999), (Radev 2000); (Mani 2001); (Maybury and Mani 2001); and (Alonso et al. 2003).

As said in the introduction, task-based competitions are one of the most important forums of discussion in the area. The TIPSTER Text Summarization Evaluation (SUMMAC) (Mani et al. 1998) and especially the DUC contest (from 2001 until today) (DUC 2007) provide a good overview of current AS systems. The Multilingual Summarization Evaluation (MSE)[1] is the special contest carried out to evaluate Arabic-English multilingual summarizers. Another relevant AS contest is the one organized by the NTCIR Japanese Project (NTCIR 2006).

The structure of this chapter is as follows. First, Section 2.1 presents a classical processing level perspective of AS, where several AS approaches are classified according to the level at which texts are analyzed. The conclusion is that the predominant tendency in AS systems is to integrate several techniques, then, a disjunct classification is difficult to establish. Section 2.2 presents new tendencies in AS from a multitask perspective. Recent state of the art multitask summarizers are classified, taking into account the kind of tasks they are able to deal with. Finally, in Section 2.3 the problem of evaluating summary quality is exposed.

2.1 Automatic Summarization from a classical perspective

There are several ways in which one can characterize different approaches to Text Summarization, such as those proposed by (Hovy and Marcu 1998; Mani and Maybury 1999; Tucker 1999; Radev 2000; Maybury and Mani 2001; Mani 2001; Alonso and Castellón 2001). In this section, we use (Mani and Maybury 1999)'s classification to describe the level at which several summarization systems process texts. Based on this classification, automatic summarizers can be characterized as approaching the problem at the surface, entity, or discourse level. The simplest type of summarizers are the ones classified at surface level, presented in Section 2.1.1. Entity level

[1] `http://projects.ldc.upenn.edu/MSE/` and `http://research.microsoft.com/~lucyv/MSE2006.htm`

approaches are discussed in Section 2.1.2 and discourse level ones in Section 2.1.3.

2.1.1 Surface level

Surface-level approaches tend to represent information in terms of shallow linguistic features that are then selectively combined together to yield a salience function used to extract information. This kind of shallow features were first used by (Luhn 1958) and (Edmundson 1969). The first one calculated sentence weights using the frequency of terms in the text and the second one taking into account sentence position in text, words occurring in the title and heading and a list of cue words.

Shallow linguistic features used in surface-level approaches can be classified as follows:

Term frequency statistics provide a thematic representation of text, assuming that important sentences are the ones that contain words that occur frequently. The salience of a sentence increases for each frequent word it contains, according to the frequencies of words in the text. The assumption is that frequent content words indicate the discussed topic, usually, in this process word stemming is used and stop words are filtered out. As said before, (Luhn 1958) proposes to extract sentences which have the highest salience value, obtaining summaries with a reasonable quality. Various frequency measures are also used by (Edmundson 1969; Kupiec et al. 1995; Teufel and Moens 1999; Hovy and Lin 1999). But the combination of word frequency with other measures does not always produce an improvement, as shown by (Kupiec et al. 1995) and (Teufel and Moens 1999). (Witbrock and Mittal 1999) obtain a statistical model to determine the likelihood that each individual word in the text has to appear in the summary.

Location relies on the intuition that important sentences are located at pre-defined positions in the text, usually genre-dependent. Instances of that assumption are the *lead method* and the *title-based method*. The *lead method* consists of taking the first sentences to create an extract-based summary. The lead method often outperforms other methods, specially with newspapers articles. The *title-based method* assumes that words in titles and headings are positively relevant to summarization. Some variants of the location method are used in (Baxendale 1958; Edmundson 1969; Donlan 1980; Kupiec et al. 1995; Teufel and Moens 1997). A generalization of location methods is the Optimum Position Policy (OPP) used by (Lin and Hovy 1997) in their SUMMARIST system, where they exploit Machine Learning (ML) techniques to identify the positions where relevant information is placed within different textual genres and domains.

Bias reflects the fact that the relevance of meaning units (sentences, clauses, etc.) is determined by the presence of terms from the title or headings, initial part of text, or user's query. On the one hand, (Kupiec et al. 1995; Teufel and Moens 1997; Hovy and Lin 1999) use those words contained in the text's title or headings. On the other hand, (Buckley and Cardie 1997; Strzalkowski et al. 1999; Hovy and Lin 1999; Conroy and O'Leary 2001; Schlesinger et al. 2002) use the words in user's query to produce query-based summaries.

Cue words and phrases are signals of relevance or irrelevance. They are typically meta-linguistic markers (e.g., cues: "in summary", "in conclusion", "our investigation", "the paper describes"; or emphasizers: "significantly", "important", "in particular", "hardly", "impossible"), as well as domain-specific bonus phrases and stigma terms. (Pollock and Zamora 1975) propose an algorithm for sentence rejection rather than selection using a list of words and a sort of marking system to decide whether a word or a phrase is an indicator for informativeness or non-informativeness. Instead, (Kupiec et al. 1995) or (Teufel and Moens 1997) use a manually built list of cue phrases. (Aone et al. 1997) detect such a list by gathering knowledge automatically from a corpus. The method started in (Teufel and Moens 1997) is expanded in (Teufel and Moens 1999), exploiting cue phrases that denote multi-sentence rhetorical blocks of text instead of single sentences.

2.1.2 Entity level

Entity-level approaches build an internal representation of the text to determine salience by modeling text entities (e.g., simple words, compound words, Named Entity (NE), terms), and their relationships (including similarity; proximity; co-occurrence relatedness; thesaural relationships among words; co-reference; and logical relations such as agreement, contradiction, entailment, and logical consistency). The first entity-level approaches based on syntactic analysis appeared in the early 1960s, (Climenson et al. 1961). Some of these approaches represent patterns of connectivity between entities in the text as a graph. Different granularities of graph nodes have been implemented (sentences (Skorochod'ko 1972), paragraphs (Salton et al. 1994) or words/phrases (Mani and Bloedorn 1997)). The underlying assumption is that highly connected nodes are more likely to carry salient information.

Relations between entities used in the internal representation of the text to determine salience include:

Similarity. Similarity between lexical units can be stated at different levels: orthographic (i.e. units sharing a common stem); distributional (i.e. units occurring in similar contexts); and semantic (i.e. units with a similar meaning). Many different approaches have been followed for measuring similarities between words or other linguistic units, ranging from vectorial distributional word based methods to semantic based ones, see (Rodríguez 2003; Budanitsky and Hirst 2006) surveys. (Myaeng and Jang 1999) use two similarity measures for determining if a sentence belongs to the main content of the document: similarity between the sentence and the rest of the document and similarity between the sentence and the title of the document. Also, in NTT (Hirao et al. 2002; Hirao et al. 2003), and CENTRIFUSER (Kan et al. 2001), a combination of similarity measures are applied.

Proximity. The distance between the text units where entities occur is a determining factor for establishing relations between entities.

Cohesion. Cohesion can be defined in terms of *connectivity*. Connectivity accounts for the fact that important text units usually contain entities that are highly connected in some kind of semantic structure. Cohesion can be approached by:

- *Word co-occurrence*: words can be related if they occur in common contexts. Some applications are presented in (Baldwin and Morton 1998; McKeown et al. 1999). (Salton et al. 1997; Mitra et al. 1997) apply IR methods at the document level, treating paragraphs in texts as documents are treated in a collection of documents. Using a traditional IR-based method, a word similarity measure is used to determine the set S_i of paragraphs that each paragraph P_i is related to. After determining relatedness scores S_i for each paragraph, paragraphs with the largest S_i scores are extracted.

  In SUMMAC (Mani et al. 1998), in the context of query-based summarization, the Smart-based approach of Cornell University and SabIR, expands the original query, compares the expanded query against paragraphs, and selects the top three paragraphs (max 25% of original) that are most similar to the original query.
- *Local salience*: important linguistic units are detected by a combination of grammatical, syntactic, and contextual parameters (Boguraev and Kennedy 1997).
- *Lexical similarity*: words can be related by thesaural relationships (synonymy, hypernymy, meronymy relations). (Barzilay 1997) details a system where Lexical Chains are used, based on (Morris and Hirst 1991). This line has also been applied to Spanish and Catalan, relying on EuroWordNet relations between words (Fuentes and Rodríguez 2002; Fuentes et al. 2004). The assumption is that important sentences are those that are crossed by strong chains. This approach provides a partial account of texts, since it focuses mostly on cohesive aspects. An integration of cohesion and coherence features of texts might contribute to overcome this, as (Alonso and Fuentes 2002) point out.
- *Co-reference*: referring expressions can be linked, and co-reference chains can be built with co-referring expressions. Both Lexical Chains and Co-reference Chains can be priorised if they contain words in a query (for query-based summaries) or in the title. In that case, the preference imposed on chain usually is: query > title > document. (Bagga and Baldwin 1998; Azzam et al. 1999) use co-reference chains for summarization. (Baldwin and Morton 1998) exploit co-reference chains specifically for query-sensitive summarization.

  The connectedness method (Mani and Bloedorn 1999) represents text with graphs in which words in the text are the nodes, and arcs represent adjacency, grammatical, co-reference, and lexical similarity-based relations.

Logical relations. Some example of this sort of relations are: agreement, contradiction, entailment, and logical consistency. (Hahn and Reimer 1999) present an approach to text summarization based on knowledge representation structures. A set of salience operators, grounded in the semantics of a terminological logic, are described and applied to knowledge databases in

order to identify concepts, properties, and relationships playing a relevant role in the text. More recently, (Lacatusu et al. 2006) propose to use Textual Entailment (TE), i.e. whether a text logically entails a summary, to better recognize relevant information across documents.

Meaning representation-based relations. An example of the relations to be established is predicate-argument, between entities in the text. (Baldwin and Morton 1998) system uses argument detection in order to resolve co-reference between the query and the text for performing summarization. (McKeown et al. 1999) use an algorithm to compare predicate argument structures of the phrases within each theme.

2.1.3 Discourse level

Discourse-level approaches model the global structure of the text, and its relation to communicative goals. Traditionally, two main properties have been distinguished in the discursive structure of a source text: cohesion and coherence. As defined by (Halliday and Hasan 1976), while cohesion tries to account for relationships among the elements of a text, coherence is represented in terms of relations between text segments, such as *elaboration*, *cause* or *explanation*. Thus, coherence defines the macro-level semantic structure of a connected discourse, while cohesion creates connectedness in a non-structural manner. In fact, approaches classified at discourse-level exploit text coherence by modeling discourse relations using different sources of evidence (document, topic, rhetorical, or narrative structure). (Correira 1980) implemented the first discourse-based approach based on story grammars (van Dijk and Kintsch 1978; vanDijk 1980).

Text coherence allows exploiting the following sources of evidence:

Format of the document (e.g., hypertext markup, document outlines) or layout (sections, chapters, etc.).

Threads of topics can be revealed in the text. An example of this is SUMMARIST, which applies Topic Identification (Hovy and Lin 1999; Lin and Hovy 2000). Topic Identification in SUMMARIST implies previous acquisition of Topic Signatures (that can be automatically learned). A Topic Signature is a set of weighted terms that characterize a topic. Then, given a text span, it is determined whether it belongs to a topic characterized by its signature. Topic Identification, then, includes text segmentation and comparison of text spans with existing Topic Signatures. Identified topics are fused during the interpretation of the process. Fused topics are then expressed in new terms. Other systems that use Topic Signatures are (Boros et al. 2001) and MEAD (Radev et al. 2000; Radev et al. 2001; Otterbacher et al. 2002). These systems assign a topic to each sentence in the text in order to create clusters to select the sentences to appear in the final summary avoiding redundancy. (Harabagiu and Lacatusu 2005) have recently extended the sets of weighted terms of Topic Signatures to include not only terms but related pairs of terms.

Rhetorical Structure of the text, representing argumentation or narrative structure. The main idea is that the coherence structure of a text can be constructed, so that the 'centrality' of the textual units in this structure will reflect their importance. A tree-like representation of texts is proposed by the Rhetorical Structure Theory (Mann and Thompson 1988). (Ono et al. 1994; Marcu 1997a) use this kind of discourse representation in order to determine the most important textual units. They propose an approach to rhetorical parsing by discourse markers and semantic similarities in order to hypothesize rhetorical relations. These hypotheses are used to derive a valid discourse representation of the original text.

2.1.4 Combined Systems

The predominant tendency in current systems is to integrate more than one of the techniques mentioned so far. Integration is a complex matter, but it seems the appropriate way to deal with the complexity of textual objects. Some examples of combination of different techniques are: (Kupiec et al. 1995; Teufel and Moens 2002; Hovy and Lin 1999; Mani and Bloedorn 1999) where title-based methods are combined with cue-location, position, and term frequency based methods.

As the field progresses, summarization systems tend to use more and deeper knowledge. For example, Information Extraction (IE) techniques are becoming widely used. Therefore, many systems do not rely any more in a single indicator of relevance or coherence, but take into account as many of them as possible. So, the tendency is that heterogeneous kinds of knowledge are merged in increasingly enriched representations of the source text(s). These enriched representations allow for adaptability of the final summary to new summarization challenges, such as MDS multilingual and even multimedia summarization. In addition, such a rich representation of text is a step forward towards generation or, at least, pseudo-generation by combining fragments of the original text. Good examples of the latter are (McKeown et al. 2002; Lin and Hovy 2002; Daumé III et al. 2002; Lal and Rueger 2002; Harabagiu and Lacatusu 2002), among others.

2.2 Automatic Summarization from a Multitask perspective

In this section the newest tendencies on AS research are presented from a multitask perspective. The different tasks proposed in the DUC contest are taken as a departure point to survey the techniques used by different systems and groups. Section 2.2.1 presents a brief description of the tasks proposed in DUC, while Section 2.2.2 proposes a classification with respect to one of the tasks that most groups tackle.

2.2.1 DUC Summarization tasks

As said previously, DUC has become the main international forum of discussion in the area of AS, with extensive evaluations carried out in several tasks since its first edition in 2001. This section briefly describes each task.

In each DUC edition, several aspects of the performance of the participant summarizers, working in the same conditions, are compared. There is no doubt that DUC contests have contributed to the development of AS systems. However, these contests are not free of critics. The main criticism is that DUC requirements are constantly changing and therefore do not allow the community research continuity.

During the two first editions of DUC contests (DUC 2001 and 2002), the summarization tasks were mainly two. A first task was the fully automatic summarization of a single newswire/newspaper document in 100 words (SDS), while the other consisted in producing single summaries given multiple newswire/newspaper documents on the same subject (MDS), with different lengths (400, 200, 100, 50 words in DUC 2001, and 200, 100, 50, 10 words in DUC 2002).

In DUC 2003, three of the four proposed tasks were new. The first task was Headline Generation (HG), consisting in producing a very short, 10-word, summary from a single document. This task was similar to one of the MDS summaries in DUC 2002, also of 10-word length and with the form of a headline. Due to the specific difficulties associated to this task in 2002, a specific task was created in 2003 to study the problem of strong compression.

The second task in 2003 was mainly the MDS task of previous DUC contests, to produce short (100-word) summaries focused on events. The third task consisted in, given a cluster of documents, producing short summaries focused on viewpoints. The viewpoint description was supposed to be a Natural Language string no larger than a sentence. It described the important facet(s) of the cluster the NIST assessor had decided to include in the short summary. These facets were represented in at least all but one of the documents in the cluster. Finally, given each document cluster, a question, and the set of sentences in each document deemed relevant to the question, the fourth task consisted in the creation of short summaries of the cluster answering that question.

In DUC 2004, two new crosslingual tasks were proposed for Arabic and English: very short crosslingual single-document summaries and short crosslingual multi-document summaries by cluster. For each task, two runs were submitted by participant. In the first run one or more automatic English translations were used, while in the second one only a manual English translation of each document was used.

In DUC 2005 only one new task was proposed: to synthesize, from a set of 25-50 documents, a brief, well-organized, fluent answer to a need for information that cannot be met by just stating

a name, date, quantity, etc. This task models real-world complex question answering and was suggested by (Amigó et al. 2004). The main difference between DUC 2005 and DUC 2006 was that in 2006 there was not indication about the granularity (specific, generic) of the desired answer, and that the questions had to be answered using less information (less documents).

In DUC 2007 two independent tasks were proposed. The main task is the same question answering task of DUC 2006. The second is a pilot task called the update task. This new task consists in producing short (100-word) multi-document update (just-the-news) summaries of newswire articles, under the assumption that the user has already read earlier information on the topic or event the articles deal with. The purpose of each update summary is to inform the reader of new information about a particular topic. The topics and documents for the update pilot task are a subset of those for the main DUC task. For each topic, documents are ordered chronologically and then partitioned into 3 sets, A-B-C, where the time stamps on all the documents in each set are ordered such that time(A) $<$ time(B) $<$ time(C). There are approximately 10 documents in Set A, 8 in Set B, and 7 in Set C.

2.2.2 Automatic Summarization Multitask classification

Initially the AS task was reduced to textual, monolingual, SDS. But it has evolved for covering currently a wide spectrum of summarization tasks that can be classified in several dimensions, among others: extracting vs abstracting, generic vs query-focused, restricted vs unrestricted domain, textual vs multimedia, SDS vs MDS, monolingual vs multilingual.

The aim of this section is to propose a generic classification of summarization tasks to reflect the research carried out in the development of systems to solve some relevant AS tasks. Figure 2.1 graphically reflects this distribution. For instance, the initial work in query-focused summarization consisted in summarizing the content of a single document to help in the detection of relevant documents when dealing with large amounts of documents, World Wide Web search engines are a good example of such systems. However, motivated by the recent DUC contest task, different techniques have been applied to MDS in order to give answer to a user need, some of which use QA systems. With respect to the language, although initially most of the work was related with single language SDS, several systems were recently applied to summarize a set of documents from different languages, some of the documents being automatic MT outputs. A MT document can be considered as ill-formed text, just as e-mails, graphics, images, multimedia, ASR or OCR output, among others. So, in Figure 2.1 it can be observed that with the exception of crosslingual summarization, until now, most of the research in summarizing ill-formed text documents involves summarizing a single document.

The tendency in recent years is to adapt the same AS approach to different tasks. As a result, there are several systems considered multitasking summarizers. More or less effort is needed when adapting a system originally designed for a specific summarization task, depending

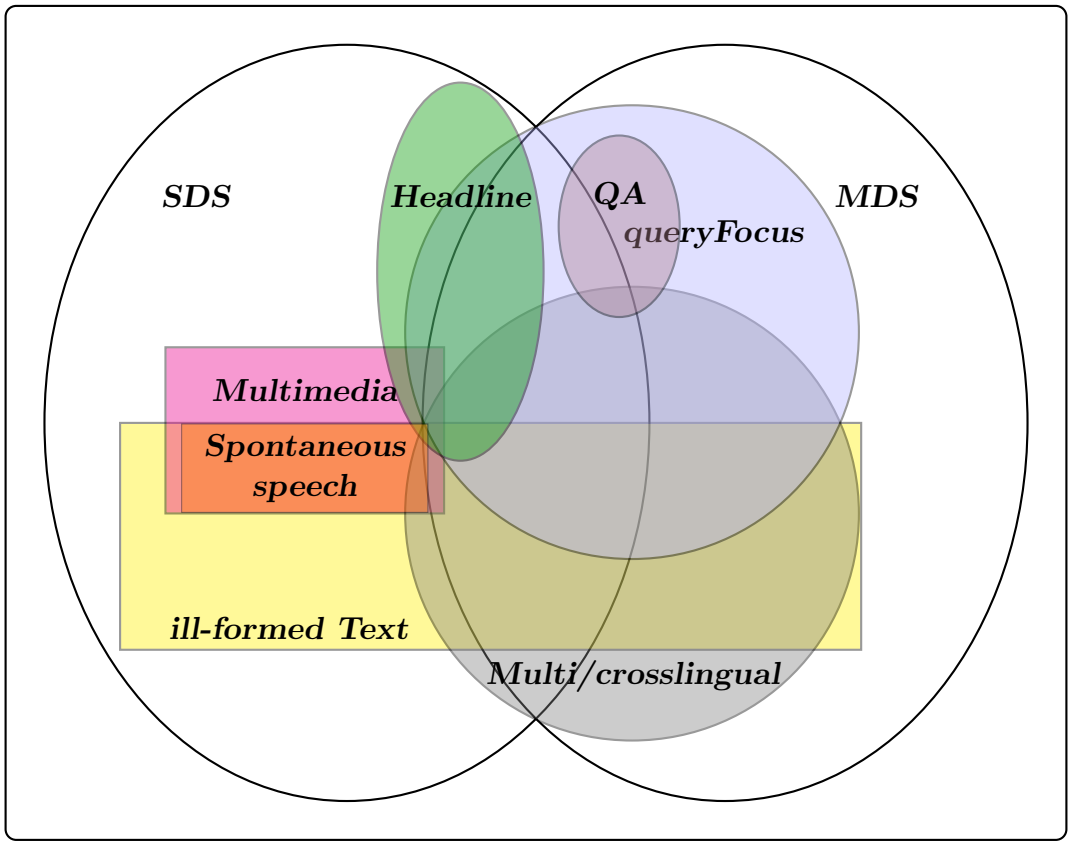

Figure 2.1: A summarization multitask classification.

not only in the tools and techniques used, but also on the characteristics of the new task. In some cases, minimal modifications are needed: That is the case of the work proposed by (Mihalcea and Tarau 2005), where a language independent SDS is used for MDS. However, many more modifications are needed when moving from Text Summarization (TS) tasks to Speech Summarization (SS) ones (Christensen et al. 2004; McKeown et al. 2005).

Another good example of an adapted system is the SumUM, originally developed by (Saggion and Lapalme 2002) to produce short automatic abstracts of long scientific and technical documents, which has been adapted to different tasks. In the first place, it was adapted by the Montreal University to face DUC 2002 SDS task (Farzindar et al. 2002), which implied a different domain, with documents of different genre and to produce shorter summaries. The following year, a new version (Farzindar and Lapalme 2003) was evaluated in the DUC 2003 MDS, focused by event task and to answer a question. After that, the SumUM system also evolved at Sheffield University. Same components were used to develop a centroid-based summarization system for DUC 2004 (Saggion and Gaizauskas 2004), a crosslingual summarization

system (Saggion 2005a), and a query-focused MDS (Saggion 2005b) one.

(Daumé III and Marcu 2006) adapted a query-focus MDS summarizer to a generic MDS one with the possibility of summarizing utomatically translated documents.

In Figure 2.1, tasks proposed in DUC or other international contests are represented by an oval shape, while rectangular shapes are used for the rest of the AS tasks. Each one of this specific tasks will be addressed below, we also present some example of the techniques used by the state-of-art AS systems to solve each task. At the end of this section some of the systems adapted to solve several DUC tasks are presented and some on-line summarizers are listed.

Single Document Summarization

Although some work on summarizing ill-formed text has been recently carried out, as exemplified in the Spontaneous Speech Summarization task description, this Section synthesizes the large amount of techniques applied to solve textual SDS. Among them, using *cohesion properties*: lexical chains, (Barzilay 1997; Brunn et al. 2001a), co-reference chains, (Baldwin and Morton 1998); *topic identification*: (Hovy and Lin 1999); *document rhetorical structure*: (Marcu 1997b; Alonso 2005); *alignment techniques*: (Banko et al. 1999); *similarity* and divergence measures, as Maximal Marginal Relevance (MMR), (Carbonell and Goldstein 1998); *statistical models*, as Bayesian models, (Schlesinger and Baker 2001), Hidden Markov Models (HMM), (Conroy et al. 2001), Logistic Regression, (Conroy et al. 2001), Latent Semantic Analysis (LSA), (Gong and Liu 2001; Steinberger and Jezek 2004; Steinberger et al. 2005), Random Indexing, (Hassel and Sjöbergh 2006); *Machine Learning* approaches, including Decision Tree (DT) and Inductive Logic Programing (ILP), (Knight and Marcu 2000; Tzoukermann et al. 2001). ML has been used mainly for two purposes: classifying a sentence from a source text into relevant or non-relevant (Kupiec et al. 1995; Aone et al. 1998; Mani and Bloedorn 1998; Lin 1999; Hirao et al. 2002) and transforming a source sentence considered relevant into a summary sentence (Jing and McKeown 2000; Knight and Marcu 2000; Harabagiu and Lacatusu 2002). *Sentence reduction* has also been applied, (Jing 2000); as well as *IE* techniques, (Kan and McKeown 1999); *graph-based algorithms*: (Erkan and Radev 2004; Mihalcea and Tarau 2005); or combinations of several techniques (Kraaij et al. 2001; Muresan et al. 2001; Alonso and Fuentes 2002).

Headline Generation

One of the last challenging AS tasks is Headline Generation (HG), also known as ultrasummarization. A headline is a highly concise representation of the most relevant points contained in a document. It can consist of a sentence, either extracted from the document or automatically generated, or, sometimes, of a list of relevant terms. The main characteristic of a headline is its extremely small length (usually between 5 and 10 words). So, in the case of extractive

HG, an aggressive condensation of the extracted sentence(s) has to be performed. Most of the techniques applied to SDS have been applied as well to HG. We can find systems ranking from purely statistical approaches to systems including different NLP tasks. However, three main subtasks are present in most HG systems:

1. *Identification of words signaling relevance.* Schiffman et al. (2001) face the identification of importance-signaling words, considering as key issues the lead property (words occurring more frequently in the lead sentences of documents), the specificity of the verbs and the use of "Concept Sets" derived from WordNet (Fellbaum 1998) instead of words. Kraaij et al. (2002) aim to identify the most informative topical Noun Phrase (NP) (for a cluster of documents). Sentences are ranked by a hybrid model that merges i) a unigram language model, mixing cluster and document models, for scoring sentences and ii) a Bayesian model on content features like cue phrases, position, length, etc. A Trigger Word Pool is built from the ranked sentences and the most salient trigger word is selected. Then, all maximal NPs containing the trigger word are found, and the one in the highest ranked sentence is taken as a headline. Daumé III et al. (2002) go beyond words or NP and perform a full parsing of the document to identify the main entities and their relations, from which the headline is built. Zajic et al. (2002) use a Noisy Channel Model (NCM), implemented by means of a HMM, for selecting a stream of headline words from a stream of story words. The first is modeled by a bigram language model and the latter by a unigram one. The headline is generated by a simple Viterbi decoder constrained by the model. A set of penalty constraints (length, position, gap and string) has been added for improving the quality of the headlines.

2. *Combination of the results proposed by simpler individual methods.* Lin (1999) uses a simple linear combination schemata with weights empirically set, while Aone et al. (1997) use DT and Nobata et al. (2001) consider different variants of 4 scoring functions (sentence position, sentence length, term frequency and distance to the title or headline). The final score of each sentence is computed by a linear combination of the individual scores. MEAD (Erkan and Radev 2004) selects one sentence as summary taking into account features such as centroid, position and length.

3. *Compression or simplification of the extracted sentences.* Knight and Marcu (2000) face the problem of sentence compression as the translation of a sentence from a source language (full text) to a target language (summary), two different approaches are followed: a NCM and a DT. Alternatively, Lal and Rueger (2002) employ lexical simplification and addition of background knowledge for modifying the initial summaries. TOPIARY (Zajic et al. 2004) uses a sentence compression parse tree approach (Dorr et al. 2003). The sentence compression is combined with the Unsupervised Topic Discovery (UTD), statistical approach to associate meaningful topic descriptors to headlines.

Multi-Document Summarization

MDS is one of the major challenges in current summarization systems. Interest in this area started with the increasing amount of on-line information. The World Wide Web is the most important source of on-line information.

The generic MDS task consists in producing a single summary of a collection of documents dealing with the same topic. This research area has been greatly shaped by the corresponding DUC task. Therefore, it has mainly focused in collections of news articles with a given topic. Remarkable progresses have been achieved in avoiding redundancy, at the beginning mainly based on the MMR proposed by Carbonell and Goldstein (1998).

Compared with SDS, new problems arise when dealing with MDS: lower compression factors implying more aggressive condensation, anti-redundancy measures, temporal dimension, more challenging coreference task (inter-document), etc. Clustering of similar documents is another new problem to be solved (Carbonell and Goldstein 1998; Radev et al. 2000; Hatzivassiloglou et al. 2001; McKeown et al. 2001). Selecting the most relevant fragments from each cluster and assuring coherence of the summaries coming from different documents are other important problems, currently under development in MDS systems.

Understanding of texts seems a desirable way of producing quality summaries. IE techniques have been applied for very restricted domains (McKeown and Radev 1995). However, systems tend to incorporate IE modules that perform a partial understanding of text, either by modeling the typical context of relevant pieces of information (Lal and Rueger 2002; Kan and McKeown 1999), or by applying general templates to find, organize and use the typical content of a kind of text or event (Harabagiu and Lacatusu 2002; Daumé III et al. 2002). This use of IE techniques produces very good results, as is reflected in the high ranking obtained by Harabagiu and Lacatusu (2002) in DUC 2002. Lin and Hovy (2002) use a combination of deeper knowledge with surface clues that seems to yield good results too.

Interesting levels of performance are also obtained applying different statistical techniques. The CLASSY system, proposed by Conroy et al. (2004), uses a HMM sentence selection approach with summary and non-summary states, co-reference resolution and sentence simplification. Saggion and Gaizauskas (2004) propose a centroid-based system where the relevance of each sentence is measured by determining the similarity to the cluster's centroid, similarity to the document lead and absolute position; a shallow redundancy detection approach, based on n-grams, is used.

The MEAD system (Radev et al. 2004; Erkan and Radev 2004) uses a graph-based algorithm to rank documents according to a linear combination of features including centroid, position and first-sentence overlap. While MEAD for MDS uses the graph-based algorithm in conjunction with other extractive summarization techniques, Mihalcea and Tarau (2005) use a graph-based algorithm alone, to summarize Single and Multiple documents. This last approach

uses a PageRank algorithm to score the nodes achieving state-of-the-art results in MDS.

As an example of a system based on ML techniques, (Hirao et al. 2003) present a similar approach to the one applied for SDS by (Ishikawa et al. 2002). Thus, an Support Vector Machine (SVM) is trained on SDS data and used to rank all the sentences from a document collection.

(Zha 2002) proposes a method for simultaneous keyphrase extraction and generic text summarization to summarize news articles and news broadcast transcripts. Documents are modeled as weighted bipartite graphs; clustering algorithms are used for partitioning sentences into topical groups. Mutual Reinforcement Learning (MRL) is used to score keyphrases or sentences from each topical group.

Query-focused Summarization

uery-focused summarization tasks are related with a specific information need expressed by the user. This need is usually expressed by a natural language query or, as in World Wide Web search engines, by a list of keywords. This kind of tasks are also known as user-focused or query-driven summarization, in contrast to generic or unbiased summarization, which is text-driven. As said in Section 1.1.1 for this task, the output summary content should be focused on the specification of the user's information need, in contrast to unbiased summarization, which should include all relevant topics.

Several works have been carried out in SDS, especially to help users detect relevant documents among large quantities of documents. A well-known example of that are Google snippets (small pieces of text that are indicative of the content of a certain link to a website). However, we are more interested in the work produced recently to solve query-focused MDS tasks.

An important step in the generation of query-focused summaries is the selection and ranking of candidate sentences. The procedures taken by most of the systems participating in the last DUC 2005 and DUC 2006 competitions combine two different kinds of metrics: one that identifies the saliency of a sentence inside a document, and another one that identifies the similarity of each sentence to the query. Some common techniques for both metrics are the following:

1. *To identify salient sentences*, it is possible to estimate the probability that a term appears in the summary by studying the term frequency in the document cluster (Nenkova and Vanderwende 2005), by an approximate oracle score (Conroy et al. 2006) or by using Bayesian approaches (Daumé III and Marcu 2005; Daumé III 2006). (Fisher and Roark 2006) train a perceptron on DUC 2001 and DUC 2002 MDS data (using ROUGE as the scoring function) to rank sentences. Other techniques are the use of centrality metrics and graph-based algorithms.

2. *To identify the similarity between the query and each sentence*, common procedures are

the use of tree similarity (Schilder and McInnes 2006), QA systems (Fuentes et al. 2005; S. Blair-Goldensohn 2006; Lacatusu et al. 2006; Molla and S. 2006) or the vector space model, possibly extended with query expansion (Alfonseca et al. 2006), syntactic and semantic relations (Lacatusu et al. 2006) or LSA and other kinds of semantic resources (Miller 2003; Jagadeesh et al. 2005; Hachey et al. 2005). As said before, (Fisher and Roark 2006) train a perceptron to rank sentences. Moreover, a second perceptron is trained on DUC 2005 to re-rank sentences taking the query into consideration.

The best system in DUC 2006 responsiveness measures, and one of the best scored in linguistic aspects, GISTexter (Lacatusu et al. 2006), combines multiple strategies for question decomposition and summary generation. In a first step, complex questions are decomposed into a set of simpler ones. The Question Processing module proceeds in three steps: keyword extraction, syntactic decomposition and semantic decomposition. The output of each of these three question decomposition strategies is sent to a QA system (Harabagiu et al. 2005) and to a MDS system, generating six candidate summaries for each complex question. Next, a textual entailmentt system is used in order to select the best candidate summary with respect to the complex question. The Modified Pyramid score (Passonneau et al. 2005) is used to score each summary.

Multilingual and Crosslingual Summarization

Regarding language coverage, systems can be classified as monolingual, multilingual, and crosslingual (a similar classification is commonly used in IR systems). Monolingual summarization systems deal with only one language for both the input document and the summary. In the case of multilingual systems, input and output languages are also the same but in this case the system can cover several languages. Crosslingual systems are able to process input documents in several languages, producing summaries also in different languages.

The main additional difficulty when dealing with multilinguality is to have the adequate language-dependent resources and tools. A more complex challenge is crosslinguality. There are examples of single document crosslingual summarizers, implying a certain amount of translation, possibly automatic, either on the input text or on the summary. Since the quality of MT is not yet good enough, it can be considered that crosslingual systems based on MT are processing ill-formed text. Nevertheless, most crosslingual summarizers are multi-document. In this case a lot of problems specific of translinguality arise. Measures of similarity between documents and passages in different languages, for identifying relations or for clustering, have to be envisaged. Similarity between lexical units (words, NEs, multiword terms) belonging to different languages, have to be computed as well.

Most multilingual systems follow statistical or ML approaches trying to be language independent. When linguistic information is used, specific processors and resources are needed for each

language and appropriate wrappers have to be built for a proper interaction between system components.

The best known multilingual summarization system is SUMMARIST (Hovy and Lin 1999). The system extracts sentences in a variety of languages (English, Spanish, Japanese, etc.) and translates the resulting summaries. SUMMARIST proceeds in three steps: topic identification, interpretation, and summary extraction. In order to face multilingual problems, the knowledge sources involved have to be as language independent as possible. In the case of SUMMARIST, sets of Topic Signatures have to be obtained for all the languages involved, using the same procedures. The segmentation procedure is also language independent. The accuracy of the resulting summaries depends heavily on the quality of the translations.

As has been said before, a more challenging issue is crosslingual multi-document. Basically, three main problems have to be addressed: 1) clustering of multilingual documents, 2) measuring the distance (or similarity) between multilingual units (documents, paragraphs, sentences, terms), and 3) automatic translation of documents or summaries. Most systems differ on the way they face these problems, the order of performance and the granularity of the units they deal with.

(Evans and Klavans 2003) present a platform for multilingual news summarization that extends Columbia's Newsblaster system (McKeown et al. 2002). The system adds a new component, translation, to the original six major modules: crawling, extraction, clustering, summarization, classification and web page generation, that have been, in turn, modified to allow multilinguality (language identification, different character encoding, language idiosyncrasy, etc.). In this system, multilingual documents are translated into English before clustering, so that clustering is performed only on English texts. Translation is carried out at two levels. A low quality translation is usually enough for clustering purposes and assessing the relevance of the sentences, that's why a simple and fast technique is applied for glossing the input documents prior to clustering. In a second step, a translation of higher quality (using Altavista's Babelfish interface to Systran) is performed only over fragments selected to be part of the summary. The system takes also into account the possible degradation of the input texts as a result of the translation process, since most of the sentences resulting from this process are simply not grammatically correct.

(Chen et al. 2003) consider three possibilities for scheduling the basic steps of document translation and clustering:

1. Translation before document clustering (as in Columbia's system), named one-phase strategy. As a result, this model clusters directly multiple documents in many languages, resulting in multilingual clusters.

2. Translation after document clustering, named two-phase strategy. This model clusters documents in each language separately and merges the results of clustering.

3. Translation deferred to sentence clustering. First, monolingual clustering is performed at document level. All the documents in each cluster refer to the same event in a specific language. Then, for generating the extracted summary of an event, all the clusters referring to this event are taken into account. Similar sentences of these multilingual clusters are clustered together, now at sentence level. Finally, a representative sentence is chosen from each cluster and translated if needed.

The accuracy of this process depends basically on the form of computing the similarity between different multilingual units. Several forms of such functions are presented and empirially evaluated by the authors. These measures are multilingual extensions of a baseline monolingual similarity measure. Sentences are represented as bags of words, only nouns and verbs are taken into account. The similarity measure is a function of the number of (approximate) matches between words and of the size of the bags. The matching function in the baseline reduces, except for NE, to the identity. In the multilingual variants of the formula, a bilingual dictionary is used as a knowledge source for computing this match. Despite of its simplicity, the position-free measure (the simplest one) seems to be the most accurate among the studied alternatives. In this approach the translations of all the words of the bag are collected and the similarity is computed as in the baseline. All the other alternatives constrain possible mappings between words in different ways, using different greedy strategies. The results of constraining mappings are, in all the cases, worse.

The two-phase strategy outperforms the one-phase strategy in the experiments. The third strategy, deferring the translation to sentence clustering, seems to be the most promising.

A system following this approach, covering English and Chinese, is presented in (Chen and Lin 2000). The main components of the system are a set of monolingual news clusterers, a unique multilingual news clusterer and a news summarizer. A central issue of the system is the definition and identification of meaningful units as the base for comparison. For English, these units can be reduced to sentences, but for Chinese the identification of units and the associated segmentation of the text can be a difficult task. Another important issue of the system (general for systems covering distant languages or different encoding schemata) is the need of a robust transliteration of names (or words not occurring in the bilingual dictionary) to assure an accurate match.

In the DUC 2004 context, to produce very short crosslingual single-document summaries, the LAKHAS system (Douzidia and Lapalme 2004) summarizes from Arabic documents and then translates. This system scores sentences taking into account the following information: lead, title, cue and Term Frequency Inverse Document Frequency (TFIDF). A sentence is reduced by name substitution, word and phrase removal.

Since the MSE contest started, several MDS approaches have been adapted to deal with crosslingual MDS. The techniques are similar to the ones used for MDS: based on clustering

(Siddhartan and Evans 2005; Dalli et al. 2006),based on a Bayesian model (Daumé III and Marcu 2006), centroid-based where the relevance of each sentence is measured by determining the similarity to the cluster's centroid (Saggion and Gaizauskas 2004; Saggion 2005a). As a ML approach, (Roark and Fisher 2005) use SVM.The MEAD system (Radev et al. 2004; Erkan and Radev 2004) uses a linear combination of features, which include centroid, position and first-sentence overlap. In a similar direction, (Wei et al. 2006) also use a graph-based ranking algorithm, but in this case nodes are scored using a PageRank algorithm. CLASSY (Conroy et al. 2005) uses a HMM for the initial score and sentence selection, while in (Conroy et al. 2006) this score is substituted by an approximate oracle score. Using a Bayesian approach, (Daumé III and Marcu 2006) model a document as a mixture of three components: a general English component, a query-specific component and a document-specific component. Sentences in a document are assigned as a continuous probability distribution of being drawn from each component. For this task, an empty string was used as query.

Speech Summarization

With the increasing importance of human-machine interaction, speech, as the most natural way of human communication, has become the core of many NLP applications. As said in Chapter 1, AS is of help in digesting the increasing amounts of information that reach us every day. Thus, it seems clear that the combination of speech and summarization, SS, will improve the interaction between humans and computers.

Leaving aside problems intrinsic to AS, the main difficulty when summarizing oral input is that the performance of ASRs is still far from perfect, and so words are often recognized wrongly (e.g.: *gate* by *Kate*), thus misleading all subsequent linguistic processing. Another specific problem is the lack of punctuation, capitalization and typographic information making more difficult all the NLP analysis, such as text segmentation, Named Entity Recognition (NER) or syntactic analysis, among others. (Zechner 2001) presents an excellent analysis of these specific problems.

As said before, extracting a summary from an input text means identifying and selecting the most relevant parts of the text, the most relevant Textual Unit (TU)s. Deciding which units are more relevant than others requires measuring the relevance of the concept represented by each of the units with respect to the set of concepts described in the whole text. However, output from ASR systems does not usually include any kind of segmentation of the recognized text into meaningful units (e.g. phrase chunks, clauses, sentences). Therefore, in order to deal with transcriptions, producing such a segmentation is a subtask of the summarization system. Statistical or ML techniques have been applied to segment dialogues into Semantic Dialogue Units or partial sentences, with or without syntactic structure. There also exists some work in monolog segmentation.

Most of SS applications have focused on Broadcast News (Jing and Hauptmann 2000; Hori et al. 2002; Stokes et al. 2004; Christensen et al. 2004; McKeown et al. 2005). Broadcast News are typically read aloud from a text. The performance of ASRs is better in read text, and the linguistic register is closer to the typical input of automatic summarizers, written text.

To produce headlines for news programs, (Jing and Hauptmann 2000) used lexical extraction methods, (Stokes et al. 2004) used lexical chains, and (Hori et al. 2002) used statistical methods to identify summary words. Such methods are all limited by the quality of the automatic transcription. To solve this problem, (McKeown et al. 2005) propose a two-level approach: first, domain specific aspects of newscasts are detected, second, portions of news stories are extracted to serve as a summary. Thus, newscasts can be searched or browsed, to locate stories of interest, and the user can ask for a summary of these stories. (Christensen et al. 2004) show that classical text summarization features are portable to the broadcast news domain. Almost all the techniques used to summarize broadcast news use lexical features derived from human or ASR transcripts features; in contrast (Ohtake et al. 2003) only use prosodic features, and (Maskey and Hirschberg 2006) exploit acoustic and prosodic features to train and test a HMM.

Besides the application to Broadcast News, Speech Summarization is also useful, for example, to grasp the main idea of political speeches or scientific conference presentations and to quickly review corporate meetings. In these cases, the problems of automatic speech recognition must be added to the problems caused by the fact that the linguistic register is very different from standard, written text: utterances are often ungrammatical or ill-formed (they contain disfluencies, repairs, or unfinished sentences), information tends to be more redundant than in written text. Despite these difficulties, there is some work on summarizing voice mail (Koumpis and Renals 2000; Koumpis and Renals 2005) utilizing prosodic features.

With respect to oral presentations, current work tends to be based on one single document, the speech transcript, (Kikuchi et al. 2003; Hirohata et al. 2005; Chatain et al. 2006), although some other work focuses on directly summarizing the speech signal of lectures (Hori et al. 2003). In this approach, recognition and summarization components are combined into a single finite state transducer.

Another spontaneous speech task is summarizing conversational speech, which is substantially different to text summarization. (Zechner 2002) summarizes spoken multiparty dialogues using MMR, including automatic speech disfluency removal, sentence boundary marking and question-answer pair detection. (Murray et al. 2005) investigate the use of several approaches: MMR, LSA, as well as a prosodic and lexical feature-based approach. (Zhu and Penn 2006) show that using more features incrementally improves the performance of summarization. In addition to using prosodic features, (Murray et al. 2006) examine the use of speech-specific characteristics such as discourse cues, speaker activity and listener feedback. In a recent work, (Galley 2006) presents different approaches for ranking by importance all utterances in a sequence using Conditional Random Fields (CRF).

Examples of Multitask Summarizers

With the aim of synthesizing the multitask summarization research, Table 2.1 presents several systems from different research groups that have been adapted to different tasks, taken from the DUC and MSE participants. The content presented in the table is as follows: In the first column, systems and research groups are identified. The rest of the columns present references related with a specific task from a specific contest. The HG task was evaluated in DUC 2002 for MDS and in DUC 2003 and 2004 for SDS. The MDS was evaluated in DUC 2002 and 2003 with well written text and in 2004 automatic MT output was included. Crosslingual evaluations were carried out in DUC 2004 and MSE 2005 and 2006. And the first query-focused summarization task was proposed in DUC 2004. While in DUC 2004 the summary should give answer to a "Who is X" question, in DUC 2005 and 2006 the query was a complex question.

Examples of on-line Summarizers

Table 2.2 lists some on-line or downloadable systems[2], both commercial and academic. For each system some information about the language and the task is provided, as well as the url were it can be found. The systems are mainly for SDS or MDS tasks, some of them potentially multilingual.

2.3 Automatic Summarization Evaluation

As said in Section 1.1.2 the evaluation of difficult NLP tasks, such AS, MT or QA has become a critical issue. The most important difficulty is that there is no single right output to any of this kind of NLP task. Extensive evaluations have been carried out as early as the 1960's (e.g. Edmundson 1969). The achieved human evaluator agreement is usually very low. Evaluation of summaries is a major issue, because objective judgments are needed to assess the progress achieved by different approaches. (Mani 2001) provides a clear picture in summary evaluation, both with human judges and by automated metrics, with a special emphasis on content-based metrics. There seems to be a general agreement on the fact that summaries are so task and genre specific, and so user oriented, that it doesn't exist a single evaluation method to cover all summarization tasks. Moreover, evaluation measures used until today are still under question.

An extensive investigation on the automatic evaluation of automatic summaries was carried out in a six-week workshop at Johns Hopkins University (Radev et al. 2002), where different evaluation metrics were proposed and the MEAD evaluation toolkit was developed. This tool consists of a large variety of evaluation metrics, including co-selection (precision, recall, F-measure, Kappa), content-based metrics (cosine, binary cosine, longest common subsequence,

[2]Site addresses valid at 6th June 2007.

System Group	HG DUC02,03,04	MDS DUC02,03,04	Crosslingual DUC04, MSE05,06	qMDS DUC04,05,06
		Schlesinger et al. 2002	Conroy et al. 2004	Conroy et al. 2004
CLASSY		Dunlavy et al. 2003	Conroy et al. 2005	Conroy et al. 2005
IDA		Conroy et al. 2004	Conroy et al. 2006	Conroy et al. 2006
		McKeown et al. 2002	Blair-Goldensohn et al. 2004	Blair-Goldensohn et al. 2004
DEMS-MG		Nenkova et al. 2003	Siddhartan and Evans 2005	Blair-Goldensohn 2005
Columbia		Blair-Goldensohn et al. 2004		S. Blair-Goldensohn 2006
	Harabagiu and Lacatusu 2002	Harabagiu and Lacatusu 2002	Lacatusu et al. 2004	Lacatusu et al. 2003
GISTexter	Lacatusu et al. 2003	Lacatusu et al. 2003		Lacatusu et al. 2005
LCC's		Lacatusu et al. 2004		Lacatusu et al. 2006
GLEANS	Marcu et al. 2002	Marcu et al. 2002		
ISI				
Marcu et al.	Daumé III and Marcu 2004	Daumé III and Marcu 2004	Daumé III and Marcu 2006	Daumé III and Marcu 2005
	Angheluta et al. 2002	Angheluta et al. 2002		
KULeuven	Angheluta et al. 2003	Angheluta et al. 2003		
	Angheluta et al. 2004	Angheluta et al. 2004		Angheluta et al. 2004
	Otterbacher et al. 2002	Otterbacher et al. 2002	Erkan and Radev 2004	Erkan and Radev 2004
MEAD		Radev et al. 2003		
UMichigan	Erkan and Radev 2004	Erkan and Radev 2004		Erkan 2006
NeATS	Lin and Hovy 2002	Lin and Hovy 2002		
ISI	Zhou and Hovy 2003			Zhou et al. 2005
Hovy et al.				
OGI/OHSU			Roark and Fisher 2005	Fisher et al. 2005
				Fisher and Roark 2006
		Farzindar and Lapalme 2003	Saggion and Gaizauskas 2004	Farzindar and Lapalme 2003
SumUM			Saggion 2005a	Saggion 2005b
USheffield			Dalli et al. 2006	
UAM	Alfonseca and Rodríguez 2003			Tobarro et al. 2005
	Alfonseca et al. 2004			Alfonseca et al. 2006

Table 2.1: Systems or Groups working in several tasks

CENTRIFUSER	English :: MDS (specific-topic: medical documents)
on-line demo	http://centrifuser.cs.columbia.edu/centrifuser.cgi
Copernic	English, French, German :: SDS (many formats)
downloadable demo	http://www.copernic.com/desktop/products/summarizer/download.html
DMSumm	English, Brazilian Portuguese :: SDS
downoadable demo	http://www.icmc.usp.br/~taspardo/DMSumm_license.htm
Extractor	English, French, Spanish, German, Japanese, Korean :: SDS (many formats)
on-line demo	http://www.extractorlive.com/
GISTexter	English :: SDS and MDS
no straightforward access	form at: http://www.languagecomputer.com/demos/summarization/index.html
GistSumm	English, Brazilian Portuguese :: SDS
downloadable demo	http://www.icmc.usp.br/~taspardo/GistSumm_license.htm
Island InText	English :: SDS
no straightforward downloading	form at: http://www.islandsoft.com/orderform.html
Inxight Summarizer /	Chinese, Danish, Dutch, English, Finnish, French, German, Italian,
LinguistX / Xerox PARC	Japanese, Korean, Norwegian, Portuguese, Spanish and Swedish :: SDS
no straightforward downloading	form at: http://www.inxight.com/products/sdks/sum/
MEAD /CLAIR	English and Chinese (multilingual) :: MDS
NewsInEssence	http://www.clsp.jhu.edu/ws2001/groups/asmd/
on-line and dowloadable demo	multiple news summ. demo at: http://www.newsinessence.com/nie.cgi
MS-Word Autosummarize	supposedly any language :: SDS
	included in MS-Word
Newsblaster	multilingual :: MDS
non-commercial research purposes	http://www1.cs.columbia.edu/nlp/newsblaster/
Pertinence Summarizer	English, French, Spanish, German, Italian, Portuguese, Japanese,
	Chinese, Korean, Arabic, Greek, Dutch, Norwegian and Russian :: SDS
on-line demo	http://www.pertinence.net
Sinope Summarizer Personal Edition	English, Dutch and German :: SDS
30-day trial downloadable	http://www.sinope.info/en/Download
Summ-It	probably English only :: pasted text
on-line demo	http://www.mcs.surrey.ac.uk/SystemQ/summary/
SweSum	English, Danish, Norwegian, Swedish,
(Web pages or pasted text)	French, German, Italian, Spanish, Greek and Farsi :: SDS
on-line demo	http://swesum.nada.kth.se/index-eng.html

Table 2.2: Some on-line demos of summarization systems, both commercial and academic.

and word overlap), relative utility, and relevance preservation. Relative Utility was tested on a large corpus in the framework of the MEAD project (Radev et al. 2003). This evaluation metric takes into account chance agreement as a lower bound and interjudge agreement as an upper bound of performance, allowing judges and summarizers to pick different sentences with similar content in their summaries without penalizing them for doing so. Each judge is asked to rank from 0 to 10 the importance of each sentence and specify which sentences subsume or paraphrase each other. Using the relative utility metric, the score of an automatic summary increases with the importance of the sentences included in it and decreases with the inclusion of redundant or irrelevant sentences.

As said in the Chapter 1, some contests have been carried out to evaluate summarization systems with common, public procedures. Specially DUC provides sets of criteria to evaluate summary quality in many different dimensions: informational coverage (precision and recall), suitability to length requirements, grammatical and discursive coherence, responsiveness, etc.

In DUC contests, from 2001 to 2004, the manual evaluation was based on a comparison with a single human-written model. DUC evaluators used the SEE software (Lin 2001) to rate the relevance of all units in the system summary by comparing them to each model unit. Much information in the evaluated summaries (both human and automatic) was marked as "related to the topic, but not directly expressed in the model summary". Ideally, this relevant information should also be scored positively during evaluation.

The early work of (Rath et al. 1961) report the disagreement between humans when perceiving relevance and (Lin and Hovy 2003) show that summaries produced by human judges are not reliable as a gold standard, because they strongly disagree with each other. As a consequence, a consensus summary obtained by applying content-based metrics, like n-gram overlap, seemed much more reliable as a gold standard against which summaries can be contrasted. They implemented the summarization package ROUGE, and proposed to use it as a method of evaluation in DUC.

For both SDS and MDS evaluations, different ROUGE measures can be used with different confidence intervals. The package is parametizable to indicate: summary target length, if Porter's stemmer has to be applied or stopwords have to be removed before computing evaluation metrics. The metrics included in the ROUGE packages are:

- ROUGE-n: n-gram co-occurrence statistics.
- ROUGE-L: Longest common substring.
- ROUGE-W: Weighted longest common substring.
- ROUGE-Sn: Skip-bigram co-occurrence statistics without gap length limit and with maximum gap lengths of n words.
- ROUGE-SUn: Extension of skip-bigram.

Although ROUGE metrics are currently used in DUC, the resulting evaluation still does not satisfactorily correlate with human assessment. Maybe with a bigger sample size correlation would improve, but extending the sample size significantly increases the cost of evaluation. Moreover, this correlation is clearly dependent on the task. Previous work (Radev et al. 2000) reports evaluation methodologies and metrics that take into account variation of content in human-authored summaries. As proposed in (Donaway et al. 2000), the tendency seems to be the study of various metrics. For example, weight n-gram matches differently according to their information content measured by Term Frequency (TF), Term Frequency Inverse Document Frequency (TFIDF), or Singular Value Decomposition (SVD).

(Amigó 2006) proposes a framework, QARLA, to combine different metrics and human or automatic judgments, in order to increase the robustness of the evaluators. Its probabilistic model is not affected by the scale properties of individual metrics. Various evaluation metrics are combined in a single measure, QUEEN. QARLA has been used for evaluating DUC 2004 tasks. This summary evaluation framework consists of three measures:

- QUEEN to determine the quality of a set of systems. The assumption underlying QUEEN is that "A good candidate must be similar to all models according to all metrics".
- KING to determine the quality of a set of metrics. KING represents the probability that, for a given evaluation metric, a human reference is more similar to the rest of human references than any automatic summary.
- JACK to determine the quality of a test set. In principle, the larger the number of automatic summaries, the higher the JACK values we should obtain. The important point is to determine when JACK values tend to stabilize, to determine when it is not useful any more to increase the number of summaries in the test set.

AutoSummENG[3] is another toolkit to evaluate automatic summaries, "peers", with respect to a set of summary models by comparing the n-gram graph representation of the peer summary to the n-gram graph representations of the model summaries. Each system is graded by a set of similarity indications (one similarity indication for each model summary on the same topic). AutoSummENG works with n-grams of words or characters.

Although reasonably good correlations with human judgements are obtained by ROUGE measures, the problem when working with n-gram models of low order is that multi-word units, such as "Single Document Summarization", are not considered as a single term. To solve that, some approaches deal with identifying semantically similar units such as *elementary discourse units* (Marcu 2000; Soricut and Marcu 2003), or the *relative utility* (Radev et al. 2000) approach, that is similar to the evaluation used by (Marcu 2000). In both cases, multiple subjects were asked to rank the importance of information units. The main difference with previous work is that the units were subsentential units rather than sentences.

[3] `http://www.ontosum.org/static/AutomaticSummarization`

In the direction of detecting elementary meaning units, two other approaches are specially remarkable. First, (van Halteren and Teufel 2003) collected 50 abstractive summaries and developed an annotation scheme for content units called *factoids*. One of the questions addressed in this work was to study consensus across summarizers to determine how many summaries are needed that obtain stable evaluation results. (Nenkova and Passonneau 2004) proposed a second approach, the Pyramid Method, an annotation method and metric that addresses the issues of interannotator reliability and the analysis of ranking stability. Within this method, a pyramid is a model to predict the distribution of summaries informative content.

To create a pyramid, annotators identify units of information found within a pool of human summaries. These basic meaning unit are refered as *Summary Content Unit (SCU)s*. An SCU is a set of contributors expressing a unique content identified by a common semantic label. These information units are usually smaller than sentences, in other NLP tasks (such as QA) have been also referred as *nuggets*. The set of words (not necessarily consecutive) from one reference summary to go into an SCU are referred to as a *contributor* of the SCU, and the shared meaning is captured in the label (see Table 2.3 for an example).

SCU LABEL: **Devolution and the Welsh assembly would happen within the first year of a Labor government**
C1: **The Labor party**, however, anxious for Plaid Cymru support, has promised to set up a Welsh assembly **within its first year of government**
C2: Labour MP candidates committed to create a Welsh "assembly", not self-government, with representation to the EC's soon-to-be-formed Committee of the Regions,**"within the life-time" of a prospective Labour-formed UK government**, as well as assemblies for the English regions, and a Scottish Parliament with taxing powers.
C3: **Within the first year** of a Labor government devolution would be expected and a Welsh assembly granted.
C4: A Labour conference promised a Welsh assembly **within the first year of a Labour government**

Table 2.3: Example of a pyramid SCU (D633.CDEH.pyr) with the **contributors** shown in their original sentential contexts.

This method addresses the summary evaluation problem by using multiple human summaries to create a summary gold standard, and exploits the frequency of occurrence of a certain piece of information in the human summaries in order to assign relative importance to different pieces

of information, represented by SCUs. Pyramids try to exploit the fact that the most relevant SCUs are those occurring in the major number of human reference summaries. The number of contributors per SCU ranges from a minimum of one to a maximum equal to the number of model summaries. SCUs with the highest number of contributors are situated at the top of the pyramid and SCUs with one contributor are situated in the lower pyramid layer.

(Nenkova 2006) studies significant differences across DUC 2004 systems as well as properties of human and system summaries. Pyramid scores analyzes summary content taking into account *SCUs*. The informativeness of a new summary can be computed based on a content pyramid. The Pyramid Method has been used to try to evaluate together content selection and linguistic quality in several DUC MDS tasks. Given a pyramid model, each peer summary produced by an automatic summarizer for a document cluster is manually annotated[4]. Table 2.4 presents several peer annotation examples[5], and Figure 2.2 shows the tool used in the process of evaluating a new summary against a pyramid (the peer summary in the left, the SCUs with its contributors in the middle, and the reference summaries in the right). The peer annotation score information appears superimposed.

In DUC 2005 was detected some disagreement problems in the pyrmaid anotation process carried out by the participants. There were few clusters with more than one manual annotation. Moreover, some inconsistencies were found, for instance system 11 obtained for cluster d324 a score of 0.2679 from one annotator and of 0.5446 from another annotator. To try to avoid inter-annotator disagreements, in DUC 2006 each set of peer summaries manually evaluated was supervised by another participant. Figure 2.3 shows an example of the kind of discussion that apperared in this process.

Two different scores are computed from peer annotations. Both scores are based on a ratio of sum of weights of the SCUs found in the peer to the sum for an ideal summary. On the one hand, the *original pyramid score* is a ratio that indicates the proportion of SCUs that were weighted with the maximum, the original score can be considered as a precision metric. On the other hand, the *modified pyramid score* ratio, a recall oriented score, measures if the summary under evaluation is as informative as one would expect given the human models.

Given the high cost of manual Pyramid annotation, the possibility of inconsistent annotations across assessors, and the additional problem of inter-annotator disagreements, a method for the automatic creation of Pyramid annotations is highly desirable. Automated approaches have been developed to match peer sentences against pyramid SCUs (Harnly et al. 2005; Fuentes et al. 2005). Both approaches still require SCUs and pyramids to be manually created a priori for reference summaries before any summary comparison can take place.

[4]The author of this thesis participated in the optional Pyramid evaluation organized by Columbia University for DUC 2005 and 2006. Participants who wished to have their automatic summaries evaluated using the Pyramid method were asked to manually annotate peer summaries' SCUs.

[5]`http://www1.cs.columbia.edu/~ani/DUC2005/`

Example A: Text in the peer that partly matches two different SCUs; note that the discriminator between the two SCUs is mainly that the second belongs to the European Community. There is some overlap in the text that matches each SCU. However, it is not the case that the exact same text matches more than one SCU.

SCU LABEL: Plaid Cymru wants full independence
Peer text: **Plaid calls for Welsh self-rule** within EC.

SCU LABEL: Plaid Cymru wants Wales to have full standing with the European Community
Peer text: **Plaid calls for Welsh self-rule within EC**.

Example B: A negative example. A too non-specific peer sentence to match any of the SCU's in the pyramid. Six of the twelve SCUs with the word assembly are listed, along with a reason why the label is too specific.

SCU Label: The Labour Party advocates an elected assembly
(Peer text doesn't mention/imply "Labour Party")

SCU Label: Labour has committed to the creation of a Welsh assembly
(Peer text doesn't mention/imply "Labour" or "commitment")

SCU Label: The British government will establish an elected Welsh assembly
(Peer text doesn't mention/imply "British government" or "establish")

SCU Label: An elected assembly would give Wales more autonomy
(Peer text doesn't mention/imply "autonomy"; SCU label doesn't mention/imply "favor")

SCU Label: Plaid Cymru's policy is to bypass an assembly
(Peer text doesn't mention/imply "Plaid Cymru" or "bypass";
SCU label doesn't mention/imply "favor")

SCU Label: The Labour Party responded to Welsh nationalist protests by favoring an elected assembly
(Peer text doesn't mention/imply "Labour Party")

Peer text: Elected Welsh assembly gains favour

Example C: A very difficult one. It requires a close reading of the model summaries to know that calls for council reform occurred in 1994, to observe that the SCU Label below matches the model text that says "postponed until at least 1995", and then to infer that the reform was to be delayed for a year, and also to infer that "the local government reform" refers to the "Welsh council reform". It is usually not necessary to do so much "research" on the pyramid and model summaries. This is a case illustrating the need for judgment: while it is a legitimate match, given the content of the model summaries, it could also be argued the other way given that the label doesn't mention "delay", and the peer doesn't mention a specifc year.

SCU LABEL: Implementation of the local government reform was scheduled for 1995.
Peer text: Welsh council reform to be delayed a year.

Table 2.4: Peer annotation examples.

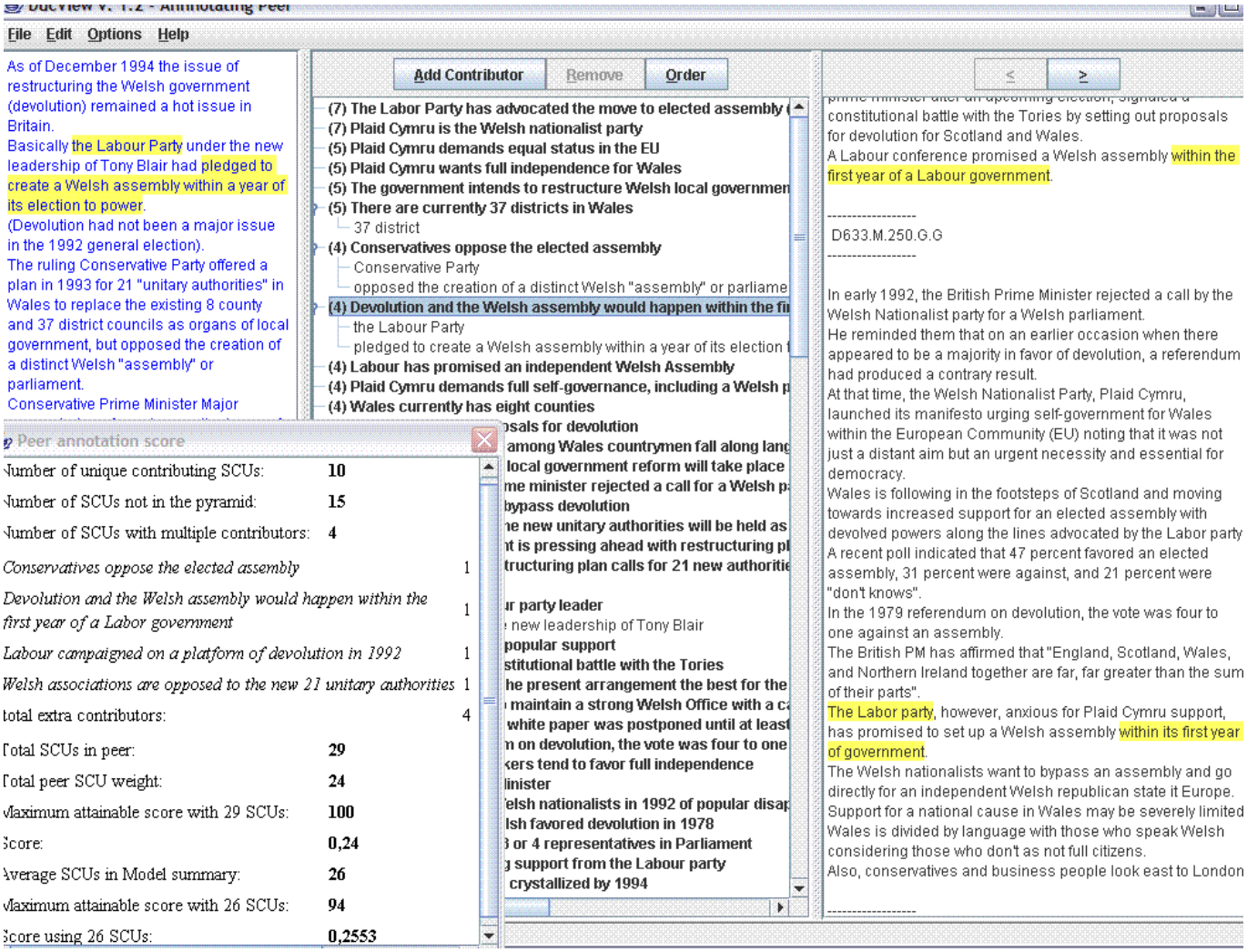

Figure 2.2: Example of an annotated peer summary, using a pyramid created from 7 human 250-word summaries.

D0631.M.250.D.6.pan

- The first sentence begins: In the aftermath of Tuesday's crash of a chartered Air France Concorde outside Paris. This implies that there are chartered Concorde flights, so I think this should match the SCU The Concorde also flew charter routes.
- The fourth sentence begins: Long an emblem of luxury and technological prowess. You have "emblem of luxury" matching the SCU The Concorde offered luxury service, including gourmet food and drink and separately "technological prowess" matching the the SCU The Concorde was considered a technological miracle when it came out. I think these are both good matches.

  I also think that the original sentence as a whole matches the SCU The Concorde was an elegant 20th Century cultural symbol. The peer doesn't actually mention the 20th century, but it does mention "long" and I think this is what is meant. It is not just separate allusions to the Concorde's luxury and technological prowess. It suggests there was a perceived golden age, which is exactly what the SCU is about.
- The same sentence continues: the supersonic jet favored by the rich and famous may slip quietly into history. I think this matches the SCU The era of Concorde flights is over. The peer doesn't say the era is over, it just questions whether it isn't over. But the original SCU contains a contributor in one of the original summaries that reads "their future use is uncertain" so I think it's close enough.

Figure 2.3: Discussion on the annotated peer summary, using a pyramid created from manual summaries.

(Hovy et al. 2005) propose grouping small, n-gram based syntactic units to automatically break down each reference summary into a minimal semantic unit, *Basic Element (BE)*. BEs are defined as the head of a major syntactic constituent (noun, verb, adjective or adverbial phrase). They try to capture lexical, semantic (based on synonymy) and distribution identity, as well as approximate match at the level of phrasal paraphrase or semantic generalization.

2.3.1 Multilingual Summarization Evaluation

In order to evaluate multilingual systems, (Saggion et al. 2002) describe a framework for the evaluation of summaries in English and Chinese using similarity measures. The framework can be used to evaluate extractive, non-extractive, single and multi-document summarization.

In the DUC 2004 contest and in the MSE editions, evaluation frameworks were defined to evaluate Arabic-English multilingual summarizers using ROUGE measures.

2.3.2 Speech Summarization Evaluation

Regarding SS evaluation, TS measures such as precision/recall and ROUGE have been used, but there are also methods designed for this specific genre, such as the Summary Accuracy in (Zechner and Waibel 2000) or the Summarization Accuracy in (Hori et al. 2003). In the oral presentation domain, (Hirohata et al. 2005) present some automatic evaluation measures (ROUGE-2, summary accuracy and F-score) to obtain high correlations with human judgments. However, in the meetings domain ROUGE scores were not found to correlate with human judgments (Murray et al. 2005; Galley 2006).

2.3.3 Query-Focused Summarization Evaluation

The QA community addresses evaluation issues related to AS when trying to evaluate answers to complex questions such as the Text REtrieval Conference (TREC) definition questions. Some common considerations in both summarization and QA communities include what constitutes a good answer/response to an information request, and one determines whether a "complex" answer is sufficient. In both communities, as well as in the new DARPA program Global Autonomous Language Environment (GALE), researchers are exploring how to capture semantic equivalence among components of different answers (nuggets, factoids or SCUs). In 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). Moreover, (Lin and Demner-Fushman 2006) incorporate judgments from multiple assessors in the form of "nugget pyramid". This work shows how to address some shortcomings with *nuggets* due to the fact that the implicit assumption that some facts are more important than others was originally implemented as a binary split between "vital" and "okay". Recently, (Zhou et al. 2007) propose a semi-automatic evaluation scheme using machine-generated nuggets.

2.3.4 International Evaluation Future

Starting in 2008, the summarization workshop will no longer be referred as DUC. Rather, the workshop will be expanded to include other NLP tasks in addition to summarization. The new workshop will be called Text Analysis Conference (TAC) and it will have multiple "tracks". TAC 2008 will have three tracks: Summarization, Question & Answering and recognizing Textual Entailment.