Summarizing Short Stories

Anna Kazantseva*
University of Ottawa

Stan Szpakowicz**
University of Ottawa
Polish Academy of Sciences

We present an approach to the automatic creation of extractive summaries of literary short stories. The summaries are produced with a specific objective in mind: to help a reader decide whether she would be interested in reading the complete story. To this end, the summaries give the user relevant information about the setting of the story without revealing its plot. The system relies on assorted surface indicators about clauses in the short story, the most important of which are those related to the aspectual type of a clause and to the main entities in a story. Fifteen judges evaluated the summaries on a number of extrinsic and intrinsic measures. The outcome of this evaluation suggests that the summaries are helpful in achieving the original objective.

1. Introduction

In the last decade, automatic text summarization has become a popular research topic with a curiously restricted scope of applications. A few innovative research directions have emerged, including headline generation (Soricut and Marcu 2007), summarization of books (Mihalcea and Ceylan 2007), personalized summarization (Díaz and Gervás 2007), generation of tables-of-contents (Branavan, Deshpande, and Barzilay 2007), summarization of speech (Fuentes et al. 2005), dialogues (Zechner 2002), evaluative text (Carenini, Ng, and Pauls 2006), and biomedical documents (Reeve, Han, and Brooks 2007). In addition, more researchers have been venturing past purely extractive summarization (Krahmer, Marsi, and van Pelt 2008; Nomoto 2007; McDonald 2006). By and large, however, most research in text summarization still revolves around texts characterized by rigid structure. The better explored among such texts are news articles (Barzilay and McKeown 2005), medical documents (Elhadad et al. 2005), legal documents (Moens 2007), and papers in the area of computer science (Teufel and Moens 2002; Mei and Zhai 2008). Although summarizing these genres is a formidable challenge in itself, it excludes a continually increasing number of informal documents available electronically. Such documents, ranging from novels to personal Web pages, offer a wealth of information that merits the attention of the text summarization community.

* School of Information Technology and Engineering, University of Ottawa, 800 King Edward Ave., Ottawa, Ontario K1N 6N5, Canada. E-mail: ankazant@site.uottawa.ca.
** School of Information Technology and Engineering, University of Ottawa, 800 King Edward Ave., Ottawa, Ontario K1N 6N5, Canada. E-mail: szpak@site.uottawa.ca.

Submission received: 3 April 2007; revised submission received: 20 January 2009; accepted for publication: 29 July 2009.

We attempt to make a step in this direction by devising an approach to summarizing a relatively unexplored genre: literary short stories.

Well-structured documents, such as news articles, exhibit a number of characteristics that help identify some of the important passages without performing in-depth semantic analysis. These characteristics include predictable location of typical items in a document and in its well-delineated parts, cue words, and template-like structure that often characterizes a genre (e.g., scientific papers). This is not the case in literature. Quite the contrary—to write fiction in accordance with a template is a sure way to write poor prose. One also cannot expect to find portions of text that summarize the main idea behind a story, and even less so to find them in the same location. In addition, the variety of literary devices (the widespread use of metaphor and figurative language, leaving things unsaid and relying on the reader's skill of reading between the lines, frequent use of dialogue, etc.) makes summarizing fiction a very distinct task. It is a contribution of this work to demonstrate that summarizing short fiction is feasible using state-of-the-art tools in natural language technology. In the case of our corpus, this is also done without deep semantic resources or knowledge bases, although such resources would be of great help. We leverage syntactic information and shallow semantics (provided by a gazetteer) to produce indicative summaries of short stories that people find helpful and that outperform naive baselines and two state-of-the-art generic summarizers.

We have restricted the scope of this potentially vast project in several ways. In the course of this work we concentrate on producing summaries of short stories suitable for a particular purpose: to help a reader form adequate expectations about the complete story and decide whether she would be interested in reading it. To this end, the summary includes important elements of the setting of a story, such as the place and the main characters, presented as excerpts from the complete story. The assumption behind this definition is this: If a reader knows when and where the story takes place and who its main characters are, she should be able to make informed decisions about it.

With such a definition of short story summaries, re-telling the plot in the summary is not among the objectives of this work; in fact, doing so is undesirable. We have introduced this limitation for two reasons. There is an "ideological" side of the decision: Not many people want to know what happens in a story before reading it, even if this may help them decide that the story is worth reading. There also is a practical side, namely the complexity of the problem: Summarizing the plot would be considerably more difficult (see Section 2 for a review of related work). We hope to tackle this issue in the future. For now, creating indicative summaries of short stories is challenge enough.

The summaries in Figures 1–3 illustrate our approach in the context of a naive lead baseline and a ceiling. Figure 1 shows an example of an automatically produced summary that meets the aforementioned criteria. A reader can see that the story is set in a restaurant where the customers are tended to by two waitresses: the fair Aileen who "wins hearts" and "the-bag-o'-meal" plain-faced Tildy. If the reader chooses to pursue the story, she will find the description of an accident of paramount importance to Tildy: One day she is kissed by a customer in public! The event is more than flattering to usually under-appreciated Tildy. It causes a complete change in how she views herself. The story then unfolds to reveal that the customer was drunk on the day in question and that he returned to apologize several days later. This apology is a severe blow to Tildy and an abrupt end of many a dream that the incident had spurred in her head. The story ends with Aileen trying to comfort her crying friend by saying "He ain't anything

THE BRIEF DEBUT OF TILDY.
By O. Henry (1862–1910).

One of the waitresses was named Aileen. She was tall, beautiful, lively, gracious, and learned in persiflage. The name of the other waitress was Tildy. Tildy was dumpy, plain-faced, and too anxious to please to please. Will it tire you to be told again that Aileen was beautiful? Tildy with the blunt nose, the hay-colored hair, the freckled skin, the bag-o'- meal figure, had never had an admirer. Tildy was a good waitress, and the men tolerated her. And Tildy was content to be the unwooed drudge if Aileen could receive the flattery and the homage. She was Aileen's friend; and she was glad to see her rule hearts and wean the attention of men from smoking pot-pie and lemon meringue.

Figure 1
Example of a summary produced by the system.

of a gentleman or he wouldn't ever of apologized." Yet, the summary in Figure 1 does not reveal these facts. For comparison, Figure 2 shows a summary obtained by taking the same number of sentences from the beginning of the story. As the reader can see, such a trivial approach is not sufficient to create a useful summary. Figure 3 shows a manually created "ideal" summary.

We experimented with a corpus of 47 stories from the 19th and early 20th century written by renowned writers, including O. Henry, Jerome K. Jerome, Anton Chekhov, and Guy de Maupassant. The stories, with the exception of a few fairy tales, are classical examples of short social fiction. The corpus was collected from Project Gutenberg (`www.gutenberg.org`) and only contains stories in English. The average length of a story is 3,333 tokens and the target compression rate expressed in the number of sentences is 94%.

In order to create summaries of short stories that satisfy our stated criteria (henceforth **indicative summaries**), the system searches each story for sentences that focus on important entities and relate the background of the story (as opposed to events). Correspondingly, processing has two stages. Initially, the summarizer identifies two types of important entities: main characters and locations. This is achieved using a gazetteer, resolving anaphoric expressions and then identifying frequently mentioned

THE BRIEF DEBUT OF TILDY.
By O. Henry (1862–1910).

If you do not know Bogle's Chop House and Family Restaurant it is your loss. For if you are one of the fortunate ones who dine expensively you should be interested to know how the other half consumes provisions. And if you belong to the half to whom waiters' checks are things of moment, you should know Bogle's, for there you get your money's worth—in quantity, at least.
Bogle's is situated in that highway of bourgeoisie, that boulevard of Brown–Jones–and–Robinson, Eighth Avenue. There are two rows of tables in the room, six in each row. On each table is a caster-stand, containing cruets of condiments and seasons. From the pepper cruet you may shake a cloud of something tasteless and melancholy, like volcanic dust. From the salt cruet you may expect nothing. Though a man should extract a sanguinary stream from the pallid turnip, yet will his prowess be balked when he comes to wrest salt from Bogle's cruets. Also upon each table stands the counterfeit of that benign sauce made "from the recipe of a nobleman in India."

Figure 2
Example of a lead baseline summary.

THE BRIEF DEBUT OF TILDY.
By O. Henry (1862–1910).

This is a story of the goings-on at a proletariat restaurant in New York City. The restaurant itself is fairly simple, two rows of six tables. Dust-like-material-filled pepper shakers and empty salt shakers decorate the tables. The owner, Bogle, sits at the cash register and makes no-nonsense change and conversation with those who eat there. Two waitresses serve the customers. Aileen, the taller, prettier, and more charming of the two is one of the reasons so many men frequent the restaurant. She flirts and attracts many regulars, who compete for her attention. Tilda, not only her co-worker, but also good friend, admires Aileen's beauty and interactions. However, while she is not jealous of Aileen's interactions with men, she does hope for someone to be infatuated with her someday.
One day, this very event takes place. Tilda receives not only a hug, but also a kiss, in public, while working! This changes her entire perception of herself, and things become even more complicated when she finds out what sparked this outburst of affection.

Figure 3
Example of a manual summary.

entities. Next, the system selects sentences that set out the background of the story and focus on one of the important entities. In order to separate the background of a story from the plot (i.e., events), we rely on the notion of aspect.[1] We approximate the aspectual type of a clause using either machine learning or manually produced rules. This is achieved by relying on an array of verb-related features, such as tense, lexical aspect of the main verb, presence of temporal expressions, and so on. Finally, the system composes a summary out of the selected sentences.

Our task remains a significant challenge despite its limited scope. To produce such indicative summaries successfully, one needs to consider many facets of the problem. An informative data representation, computational complexity, and usability of the final product are only some of them. Because the project is at the stage of an advanced feasibility study, it has not been possible to do justice to all aspects of the problem. Therefore, we concentrated on several specific issues and left many more to future work and to fellow researchers.

Firstly, we sought to identify characteristics of short stories that could be helpful in creating summaries. We devised an informative and practical data representation that could be reproduced without too much cost or effort. Secondly, we restricted ourselves to identifying the most informative portions of the stories and paid much less attention to readability and coherence of the resulting summaries. Even though readability is an important property, we hypothesized that informativeness is yet more important. Once the task of identifying informative passages has been accomplished, one can work on achieving coherence and readability. In the end, the emphasis was on the creation of extractive summaries using established tools and methods and on the identification of genre-specific properties that can help summarization.

The novelty of the task and the absence of agreed-upon measures for evaluating summaries of literary prose call for a thorough evaluation using a variety of metrics. That is why we conduct three distinct evaluation experiments. The summaries are

1 The term **aspect** is defined and explained in detail in Section 4. For now it suffices to say that by aspect we mean a characteristic of a clause that gives readers an idea about the temporal flow of an event or a state described in it. For example, the aspectual type of the sentence *He likes to run* is a *state*. The aspectual type of *He has run a marathon* is an *event*.

evaluated both extrinsically and intrinsically. They are also compared with two naive baselines (lead and random) and two state-of-the-art summarization systems designed for summarizing newswire (henceforth **baseline summarizers**).[2]

In the first experiment, 15 people read a mix of machine-made, random, and manual summaries, and answer questions about them. Some questions are factual in nature (e.g., *list the main characters of the story*), and others are subjective (e.g., *rate the readability of the summary*). The results show that the machine-made summaries are significantly better than the random baseline but they fall far short of the quality of the manual summaries.

During the second evaluative experiment, the machine-made summaries are compared against extracts created by people using sentence co-selection measures (precision, recall, and F-score). By sentence co-selection we mean measuring how many sentences found in manually created extracts are selected for inclusion in automatically produced summaries. The results suggest that our system outperforms all baselines, including state-of-the-art summarizers.

The third part of the evaluation uses two ROUGE metrics (Lin 2004) to compare the machine-made and the baseline summaries with the model abstracts. The results suggest that these measures are not well suited for evaluating extractive indicative summaries of short stories.

This paper is organized in the following manner. Section 2 gives a brief overview of the body of research in automatic story comprehension. Section 3 describes the process of identifying important entities in short stories. Section 4 introduces the notion of aspect, gives an overview of the system's design, and discusses the linguistic motivation behind it. Section 5 describes the classification procedures (the use of machine learning and manual rule creation) that distinguish between the descriptive elements of a story and the passages that describe events. Section 6 reports on the evaluation of summaries which our system produces. Section 7 draws conclusions and outlines directions for future work.

2. Related Work

Summarization of literary prose is a relatively unexplored topic. There exists, however, a substantial body of research tackling the problem of story comprehension. During the 1970s and 1980s, a number of researchers in artificial intelligence built story-understanding systems that relied in one way or another on contemporary research in psychology and discourse processing.

Much of that line of research relied on an assumption that stories exhibit global cognitive structure (known as **macrostructure** [van Dijk 1980] or **schema** [Bartlett 1932]) and that they can be decomposed into a finite number of cognitive units. According to this view, diversity in stories is not due to an infinite number of plots, but to an infinite number of combinations of a (relatively) small number of cognitive units. This direction was pioneered in 1928 by Vladimir Propp (1968) with his detailed analysis of 100 Russian folk tales. After a period of oblivion, these ideas regained popularity: van Dijk and Kintsch (1978) demonstrated the existence of macrostructures and their role in story comprehension and recall; Rumelhart (1975), Thorndyke (1975), and Mandler

2 These systems are GISTexter (Harabagiu, Hickl, and Lacatusu 2007) and CLASSY (Schlesinger, O'Leary, and Conroy 2008; Conroy, Schlesinger, and O'Leary 2007). See Section 6 for details.

(1987) developed sets of cognitive schemas (story grammars) that could be applied to certain types of stories; and Lehnert (1982) proposed to represent action-based stories in terms of a finite number of plot units—configurations of affect (or emotional) states of its characters.

The developments in psychology and discourse processing had gone hand in hand with those in artificial intelligence and resulted in a score of story-understanding and question-answering systems. Many of these relied on a set of manually encoded schemas and chose the most appropriate one for a given story (Cullingford 1978; Dyer 1983; Leake 1989). For example, a system called BORIS (Dyer 1983) processed stories word-by-word to create a very rich semantic representation of them using Memory Organization Packets (MOPs) and Thematic Affect Units (TAUs). These knowledge structures were activated by means of a very detailed lexicon where each lexeme was associated with MOPs and TAUs it could invoke.

Systems such as BORIS could not process stories that did not conform to schemas already at their disposal. Charniak and Goldman (1988) and Norvig (1989) attempted to circumvent this problem by learning to recognize more general structures. FAUSTUS (Norvig 1989) recognized six general classes of inferences by finding patterns of connectivity in a semantic network. It could be adapted to new kinds of documents by extending its knowledge base and not the underlying algorithm or patterns. Research in automatic story comprehension offered a number of important solutions for subsequent developments in artificial intelligence. No less important, it pointed out a number of challenges. All these systems required a formidable amount of semantic knowledge and a robust and efficient way of building a semantic representation of texts. In addition, systems such as BORIS or SAM (Cullingford 1978) also needed a set of schemas or schema-like scenarios. With such labor intensity, these requirements prohibit using schema-based approaches for real-life stories (e.g., fiction) and only allow the processing of artificially created examples.

In this historical context, our current approach to summarization of short fiction appears rather modest: Our system does not "understand" stories, nor does it retell their plot. Instead, it offers the reader hints—important information about the story's setting—which should help her guess what type of story is to come. This assumption appears reasonable because it has been shown that comprehension and recall of discourse are strongly influenced by the reader's familiarity with the type of schema (van Dijk and Kintsch 1978). Because our system is tailored to work with classics of the genre, it was our expectation that the gist of the story's setting offered to the reader in the wording of the original would give her an idea about the story's themes and likely plot developments. The results of our experiments appear to back this assumption.

In addition, given the original objective, it seems reasonable that elements other than the plot would have a strong influence on the reader's decision to read or not to read the story. The setting of the story, its characters, and style are some of the important factors. Many outstanding literary works differ not so much in plot as in their setting or moral.[3] Our system attempts to capture important elements of the setting explicitly and we expect that some elements of style may be captured implicitly due to the extractive nature of the summaries.

3 Consider, for example, Goethe's *Faust* and Bulgakov's *Master and Margarita*. Although they both revolve around the protagonist entering into a pact with the devil—albeit for different reasons—the latter takes place in Moscow around 1930s and the two works are dramatically different.

3. Identifying Important Entities

During the first stage of summary production the system identifies important entities in stories. Initially, we planned to identify three types of entities: people, locations, and time stamps. During a preliminary exploration of the corpus, we analyzed 14 stories for the presence of surface indicators of characters, locations, and temporal anchors.[4] We employed the GATE Gazetteer (Cunningham et al. 2002), and only considered entities it recognized automatically.

The experiment revealed that the stories in the corpus contained multiple mentions of characters (on average, 64 mentions per story, excluding pronouns). On the other hand, the 14 stories contained only 22 location markers, mostly street names. Four stories had no identifiable location markers. Finally, merely four temporal anchors were identified in all 14 stories: two absolute (such as year) and two relative (e.g., Christmas). These findings support the intuitive idea that short stories revolve around their characters, even if the ultimate goal is to show a larger social phenomenon. They also suggest that looking for time stamps in short stories is unlikely to prove productive, because such information is not included in these texts explicitly. That is why our system does not attempt to identify them.

Because characters appear to be central to short stories, we designed our system to maximize the amount of information available about them. It contains an anaphora resolution module that resolves pronominal and noun phrase anaphoric references to animate entities. The term **anaphora**, as used in this work, can be explained as a way of mentioning a previously encountered entity without naming it explicitly. Consider Examples 1a, 1b, and 1c from "A Matter of Mean Elevation" by O. Henry. The noun phrase *Mlle. Giraud* from Example 1a is an **antecedent** and the pronouns *her* and *she* from Example 1c are **anaphoric expressions** or **referents**. Example 1c illustrates **pronominal anaphora**, and Example 1b illustrates **noun phrase anaphora**. Here the noun phrase *the woman* is the anaphoric expression which refers to the antecedent *Mlle. Giraud* from Example 1a.

(1a) *John Armstrong*$_{ent1}$ and *Mlle. Giraud*$_{ent2}$ rode among the Andean peaks, enveloped in their greatness and sublimity.

(1b) To *Armstrong*$_{ent1}$ *the woman*$_{ent2}$ seemed almost a holy thing.

(1c) Never yet since *her*$_{ent2}$ rescue had *she*$_{ent2}$ smiled.

The anaphora resolution module only handles first and third person singular personal pronouns (*I, me, my, he, his*...) and singular definite noun phrases that denote animate entities (e.g., *the man*, but not *men*). It is implemented in Java, within the GATE framework, using the Connexor Machinese Syntax parser (Tapanainen and Järvinen 1997).

The system resolves anaphoric expressions in the following manner. Initially, the documents are parsed with the Connexor Machinese Syntax parser. The parsed data are then forwarded to the Gazetteer in GATE, which recognizes nouns denoting locations and persons. The original version of the Gazetteer only recognizes named entities and professions, but we extended it to include 137 common animate nouns such as *man*, *woman*, *soldier*, or *baby*. During the next stage, pronominal anaphoric expressions are

4 The stories used in this exploration were later included in the training part of the data. They were never used for testing.

Table 1
Results of anaphora resolution.

Type of anaphora	All	Correct	Incorrect	Error rate, %
Pronominal	597	507	90	15.07
Nominal	152	96	56	36.84
Both	749	603	146	19.49

resolved using an implementation of the algorithm proposed by Lappin and Leass (1994).[5] Subsequently, anaphoric noun phrases are identified using the rules outlined by Vieira and Poesio (2000). Finally, anaphoric noun phrases are resolved using a modified version of the Lappin and Leass algorithm, adjusted to finding antecedents of nouns. The implementation is described in detail in Kazantseva (2006).

A thorough evaluation of the anaphora resolution module would be prohibitively labor-intensive. We estimated the performance of the module by manually verifying the results it achieved on two short stories of the training set (Table 1). The error rates for pronominal anaphora resolution are significantly lower than those for noun phrase anaphora resolution (15.07% vs. 36.84%). This is not unexpected because resolving noun phrase anaphora is known to be a very challenging task (Vieira and Poesio 2000). The results also reveal that referring to characters by pronouns is much more frequent than by noun phrases—in our case, the ratio of pronominal to nominal expressions is almost 4:1. This suggests that resolving pronominal anaphoric expressions is crucial to summarizing short stories.

The GATE Gazetteer, part of this module, also annotates the stories for the presence of expressions denoting locations. After resolving anaphoric expressions, characters central to each story are selected based on normalized frequency counts taking anaphoric expressions into account. The output of this module consists of short stories annotated for the presence of location markers and main character mentions.

4. Selecting Descriptive Sentences Using Aspectual Information

4.1 Linguistic Definition of Aspect

We rely on aspect to select salient sentences that set out the background of a story. In this paper, the term **aspect** denotes the same concept as what Huddleston and Pullum (2002, page 118) call the **situation type**. The term refers to "different ways of viewing the internal temporal consistency of a situation" (Comrie 1976, page 3). Informally, the aspect of a clause suggests the temporal flow of an event or a state and the speaker's position with respect to it.

A general aspectual classification based on Huddleston and Pullum (2002) appears in Figure 4, with examples for each type.

5 Lappin and Leass (1994) present a rule-based algorithm for resolving pronominal anaphora. The algorithm suggests the most likely antecedent after taking into account the candidates' syntactic function, recency, and absence or presence of parallelism and cataphora with the referent. It also enforces agreement between referent–antecedent pairs.

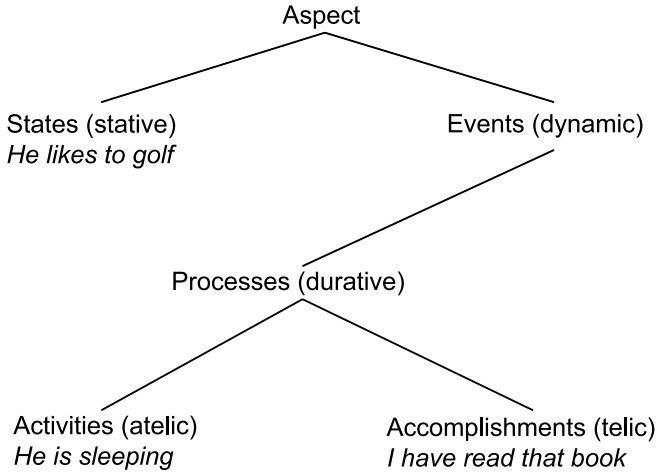

Figure 4
Aspectual hierarchy after Huddleston and Pullum (2002).

The first distinction is between **states** and **events**. Events are processes that go on in time and consist of successive phases (Vendler 1967, page 99). For instance, an event of writing an essay consists of writing separate words, correcting, pausing between words, and so on. A state of understanding each other, on the other hand, does not imply such compositionality: It remains unchanged throughout the whole period when it is true. In other words, the meaning of events exhibits a dynamic component, whereas that of states does not.

Events are further categorized by whether a particular situation lasts for some time or occurs momentarily. The latter type of events are referred to as **achievements**, and events that imply duration are known as **processes**. For example, the nature of events such as dropping, stumbling, or recognizing is that they occur instantaneously and, therefore, are achievements. On the other hand, events such as playing golf or writing an essay last for some time, so they are processes.

Processes are classified into **accomplishments** and **activities** depending on whether a situation implies an ending (Vendler 1967, page 100). This property is known as **telicity**. Reading a book in the context of Example 2a implies that the person finished reading it: the overall situation is **telic**. We cannot say that she has read the book in the first 15 minutes of doing so because the implied ending was not achieved (i.e., the book has not been read). Such situations are referred to as **accomplishments**. On the other hand, playing golf or talking on the phone does not imply that the process must end with a specific conclusion and the situation is **atelic**. Such situations are called **activities**.

In addition, the aspectual type of a clause may be altered by multiplicity, for example, repetitions. Consider Examples 2a and 2b.

(2a) She read a book.

(2b) She read a book a day.

Example 2b is referred to as a **serial situation** (Huddleston and Pullum 2002, page 123). It is considered to be a state, even though a single act of reading a book would constitute an event.

Intuitively, stative—and especially serial—situations are more likely to be associated with descriptions, that is to say, with things that are, or things that were happening for an extended period (consider *He was a tall man* vs. *He opened the window*). The remainder of Section 4 describes how we identify single and serial stative clauses and use them to construct summaries.

4.2 Overall System Design

Several system components are responsible for selecting salient background sentences. A story, annotated for the presence of important entities (as outlined in Section 3), is parsed with the Connexor Machinese Syntax parser. The sentences are then recursively split into clauses based on the results of parsing. For the purposes of this project, a **clause** is defined as a main verb as identified by the parser (whether finite or non-finite) with all its complements, including subject, modifiers, and their constituents.

Next, each clause is represented as a vector of features describing its characteristics. The system offers a choice: a fine-grained or coarse-grained representation. The main difference between the two is in the level of detail at which each clause is represented. For instance, a fine-grained feature vector has three different features with seven possible values to carry tense-related information: *tense*, *is_progressive*, and *is_perfect*, whereas a coarse-grained vector carries only one binary feature, *is_simple_past_or_present*.[6]

Finally, the system selects salient descriptive sentences. Regardless of the granularity of the representation, one may choose between two different procedures for sentence selection. The first procedure employs machine learning techniques, namely the C5.0 decision tree induction (Quinlan 1992). The second procedure applies a set of manually created rules that guide the classification process. Section 4.3 gives a motivation for features used in each data set. Sections 5.1–5.3 describe the experimental setting. Section 6 presents the results.

The part of the system that selects descriptive sentences is implemented in Python.

4.3 Feature Selection: Description and Motivation

There are two criteria for the selection of features for both representations:

(Criterion 1) a clause should "talk" about important things, such as characters or locations

(Criterion 2) a clause should contain background descriptions rather then events

We hypothesize that sentences which satisfy both criteria are good candidates for inclusion in indicative summaries. In other words, a summary that consists of such sentences would familiarize the reader with important elements of the setting of the story, but would not reveal the plot.

The features that contribute towards Criterion 1 can be divided into character-related and location-related. We have designed character-related features to help identify sentences that focus on characters, not just mention them in passing. These features are modeled so as to help identify sentences that contain at least one mention of an important character with a salient grammatical function (e.g., subject). Location-related

6 Furthermore, in this article we refer to a data set annotated with the fine-grained features as the **fine-grained data set**, and to the one annotated with the coarse-grained features as the **coarse-grained data set**.

Table 2
Description of the features in both data sets.

	Fine-grained data set		Coarse-grained data set	
Type of features	Number of features	Number of values	Number of features	Number of values
Character-related	10	18	4	6
Aspect-related	14	48	6	15
Location-related	2	4	2	4
Other	3	7	3	4
All	29	77	15	29

features are intended to help identify sentences where named entities tagged as locations by the Gazetteer indeed refer to location names.

Criterion 2 has been introduced to ensure that the selected sentences are background sentences (as opposed to those relating events) and are therefore suitable for inclusion in indicative summaries. To this end, the features that contribute towards Criterion 2 are designed to identify stative clauses and clauses that describe serial situations. A single unambiguous indicator of aspectual type does not exist, but a number of verb-related characteristics of the clause may signal or limit its possible aspectual type. These characteristics include the lexical aspect of the main verb, tense, the presence of temporal expressions, voice, and certain properties of the direct object. The verb-related features capture this information in our representation.[7]

The remainder of this section contains a detailed description of the various types of features and motivates their inclusion. Table 2 shows how many features contribute to each criterion, and how many discrete values they have. Appendix A contains a complete list of features used in both representations, explains how they are computed, and shows the cardinality of the sets of possible values.

Character-related features. Character-related features help ensure that selected sentences are about one of the important characters in the story. So, this group of features describes whether a clause contains a character mention and what its grammatical function is (subject, object, indirect object, or other). Mentions of characters early in the text tend to contain more salient background information. That is why character-related features reflect the position of a parent sentence[8] relative to the sentence where the character is introduced. In addition, these features capture the presence of a character mention that is premodified by a noun phrase. The interest in such mentions is inspired by the fact that these constructions—appositions—often introduce new entities into the discourse (Vieira and Poesio 2000). For the same reasons, the system also establishes whether a character mention is nominal or pronominal (e.g., *Jack* vs. *he*), whether it is used in the genitive case (e.g., *Jack's*) and, for common nouns, whether the mention is accompanied by an indefinite article.

7 It must be mentioned that several researchers have looked into automatically determining various semantic properties of verbs (Siegel 1998b; Merlo et al. 2002). These approaches, however, attempt to determine properties of verbs viewed in isolation and do not deal with particular usages in the context of concrete sentences. That is why we cannot directly re-apply that research in determining the aspectual type of clauses.

8 By **parent sentence** we mean the sentence from which the clause is taken.

Table 3
Privative featural identification of aspectual classes after Dorr and Olsen (1997).

Aspectual class	Telic	Dynamic	Durative	Examples
State			+	know, believe
Activity		+	+	paint, walk
Accomplishment	+	+	+	destroy
Achievement	+	+		notice, win

Location-related features. In discourse such as fiction, not all tokens that the Gazetteer recognizes as markers of location denote locations. Location-related features help identify mentions of locations in each clause and verify that these mentions indeed denote a place. These features describe whether a clause contains a mention of a location and whether it is embedded in a prepositional phrase. The rationale for these features is that true location mentions are more likely to occur inside prepositional phrases, such as *from Chicago* or *to China*.

Verb-related features. Verb-related features model the characteristics of a clause that help determine its aspectual type.

Lexical aspect of a verb. Lexical aspect refers to a property of a verb when viewed in isolation, without regard to the context provided by a particular clause. Just as for clauses, a verb may be a state (or stative) verb (e.g., *believe*), or an event verb (e.g., *run*). Event verbs are further subdivided into verbs of activity (e.g., *read*), accomplishment (e.g., *take a test*), and achievement (e.g., *drop*).

The relation between the lexical aspect of a verb and the aspect of a clause has been discussed by Vendler (1967), Dorr and Olsen (1997), and Huddleston and Pullum (2002, pages 118–123). Dorr and Olsen have proposed a privative model of this relation—see Table 3. The model states that verbs are categorized into aspectual classes based on whether they exhibit one or more of the following properties: dynamicity, durativity, and telicity. Dorr and Olsen speculate that, depending on the context of usage, verbs may form clauses that have more of these properties than the main verb viewed in isolation, but that it is impossible for a verb to "shed" one of its properties. We illustrate this in Examples 3a and 3b. In Example 3a the state verb *know* participates in an accomplishment clause; the clause is telic, although the verb by itself is not. On the other hand, an attempt to deprive the accomplishment verb *destroy* of its telic meaning when constructing a clause of type *activity* fails to create an acceptable clause (Example 3b).

(3a) He knew it that very moment. (accomplishment)

(3b) *He was destroying it for an hour. (activity) [9]

The lexical aspect of a verb influences the aspect of a clause. Several features in our system capture this information. The fine-grained data set contains three features with six possible values that show whether the main verb of a clause is durative, dynamic, or telic. The coarse-grained data set contains a single feature with four possible values (the lexical aspect of a verb according to the model in Table 3). We derive this information from a manually compiled database of Lexical Conceptual Structures (Dorr and Olsen 1997), which contains these properties for 4,432 English verbs.

9 Throughout this paper, the asterisk (*) denotes incorrect or marginally correct usage.

Grammatical tense. The grammatical tense used in a particular clause places a number of constraints on its aspectual type. For instance, simple tenses are more likely to be used in stative or habitual situations than progressive or perfect tenses. It is also commonly accepted (Dowty 1979; Huddleston and Pullum 2002, page 119) that stative clauses cannot be realized using progressive tenses (see Examples 4a and 4b). Huddleston and Pullum (2002, page 121) stipulate that it is also the case with achievement clauses (see Example 4c).

(4a) John is running. (event, activity)

(4b) *John is knowing the answer. (state)

(4c) *John was recognizing her. (event, accomplishment)

Among the constraints that grammatical tense imposes there is the special relation between simple present tense and event clauses. As a rule, clauses realized in simple present tense cannot denote events, but only states (Huddleston and Pullum 2002, page 119). The matter is illustrated in Examples 5a through 7b.

(5a) She knew history well. (state)

(5b) She knows history well. (state)

(6a) She fell off a chair. (event)

(6b) *She falls off a chair. (event)

(7a) She danced (last night). (event)

(7b) She dances. (state)

In the fine-grained data set the information related to tense is expressed using three features with seven possible values (whether a clause is in present, past, or future tense; whether it is progressive; and whether it is perfective). In the coarse-grained data set, this information is expressed using one binary feature: whether a clause is in simple, past, or present tense.

Temporal expressions. Temporal markers (often referred to as temporal adverbials), such as *usually, never, suddenly, at that moment*, and many others, are widely employed to mark the aspectual type of a sentence (Dowty 1979; Harkness 1987; By 2002). Such markers provide a wealth of information and often unambiguously signal aspectual type. For example:

(8a) She read a lot tonight.

(8b) She always read a lot. (or She used to read a lot.)

Such expressions are not easy to capture automatically, however. In order to use the information expressed in temporal adverbials, we analyzed the training part of the corpus for the presence of such expressions. There were 295 occurrences in 10 stories. It turns out that this set can be reduced to 95 templates. For example, the expressions *this year, next year, that long year* can all be reduced to a template ⟨*some_expression year*⟩. Possible values of ⟨*time_expression*⟩ are further restricted to allow only valid modifiers (e.g., *last*, *next*, but not *yellow*). The system captures temporal expressions using a cascade of regular expression. It first identifies the least ambiguous unit (in this example

year) and then attempts to find the boundaries of the expression. The complete list of regular expressions used appears in Kazantseva (2006).

Three features characterize each template: the type of the temporal expression (location, duration, frequency, or enactment) (Harkness 1987); magnitude (year, day, etc.); and plurality (year vs. years). The fine-grained data set contains three such features with 14 possible values (type of expression, its magnitude, and plurality). The coarse-grained data set contains one binary feature (whether a clause contains an expression that denotes a long period of time).

Voice. Usually, clauses in passive voice only occur with events (Siegel 1998b, page 51). Both data sets contain one binary feature that describes this information.

Properties of the direct object. For some verbs, properties of the direct object help determine whether a given clause is stative or dynamic.

(9a) She wrote a book. (event)

(9b) She wrote books. (state)

It is of particular interest whether the direct object follows a definite or indefinite determiner and whether it is used in a singular or plural form. Two binary features that describe this information are included in the fine-grained data set.

Several additional features in both data sets describe the overall characteristics of a clause and its parent sentence, such as whether these were affirmative statements, exclamations, or questions; their index in the text; and a few others. The fine-grained data set contains three such features with seven possible values. The coarse-grained data set contains three features with four values.

4.4 Handling Clauses with the Verb *Have*

The preceding section notes that the same verb may form clauses of different aspectual types depending on its context. A verb with a particularly ambiguous aspect is the verb *have* (when used as the main verb and not an auxiliary). Its meaning is strongly influenced by what kind of direct object it takes. That is why determining its aspectual type is a very challenging task. This issue is illustrated in Examples 10a–10c.

(10a) She had lunch. (event).

(10b) She had a friend. (state).

(10c) She had an accident. (event).

Due to the high degree of ambiguity, our system handles clauses with *have* as the main verb in a manner different from all other clauses. This machinery remains the same regardless of what options are used for the granularity of representation and for sentence selection procedures.

In order to handle *have*-clauses, our system contains an implementation of an approach proposed by Siegel (1998a). The solution relies on WordNet (Fellbaum 1998) and contains a set of rules that determine the aspectual type of a *have*-clause based on the top WordNet category of the direct object. For instance, the direct object *lunch* from Example 11a belongs to the category *food* and, according to rules from Siegel (1998a), the aspectual type of a clause is *event*. The direct object *friend* from Example 11b belongs to the category *person*, so the aspectual type of the clause is *state*. Siegel (1998a) used WordNet 1.6, whereas we work with a newer version, WordNet 2.0. The structure of

stateCategories = ['cognition', 'state', 'time', 'artifact', 'attribute', 'entity', 'measure', 'substance', 'relation', 'person', 'group', 'location', 'feeling', 'pronoun', 'animal']

if parent categories of the hypernym tree contain at least one of the *stateCategories*:
    return True //stative clause
else:
    return False //dynamic clause

Figure 5
Pseudo-code for determining the type of *have*-clauses based on the WordNet category of direct object (Siegel 1998b).

this newer ontology is different from that of version 1.6. For this reason, we consider all parent categories in the hypernym tree, not only the top category. For the sake of completeness, Figure 5 shows the pseudo-code for this procedure. The system judges a *have*-clause to be summary-worthy if two conditions are fulfilled: the clause contains a mention of one or more important characters and it is a state clause.

5. Experiments

5.1 Experimental Setting

The final version of the summarizer proceeds as follows. First of all, the stories are parsed with the Connexor parser and named entities are recognized using the GATE Gazetteer. Then the system resolves anaphoric references and identifies important characters and locations. During the next stage, the summarizer splits all source sentences into clauses and creates coarse- and fine-grained representations for each clause. A clause is modeled as a vector of character-, location- and verb-related features. Finally, the system employs two alternative procedures to select summary-worthy sentences: manually designed rules and machine learning.

We performed a number of experiments to find out how successful our system is in creating summaries of short stories. The experimental corpus consisted of 47 short stories split into a training set of 27 stories and a test set of 20 stories. The average length of a story in the corpus was 3,333 tokens, 244 sentences, or approximately 4.5 U.S.-letter-sized pages. The corpus contains stories written by 17 different authors. It was split manually so that its training and test portions contained approximately an equal number of stories by the same writer. The first author of this paper annotated each clause of every story for summary-worthiness and achieved the compression rate of 6%, counted in sentences. This rate was the target compression rate in all further experiments.

The training data set consisted of 10,525 clauses, 506 of which were annotated as summary-worthy and all others as not summary-worthy. The test data set contained 7,890 clauses, 406 of them summary-worthy.

We fine-tuned the system and used the training portion of the data set to identify the best settings. Then we ran two sets of experiments on the test portion. In the first set of experiments, we applied a manually designed set of rules that select sentences for possible inclusion in summaries. These experiments are described in Section 5.2. The second set of experiments relied on using machine-learning techniques to create summaries. It is described in Section 5.3. After the completion of the experiments, the summaries were evaluated by six judges. They were also compared against

Rule 1
if a clause contains a character mention as subject or object **and** a temporal expression of type *enactment* (e.g., *ever, never, always*)
    return True
Rule 2
if a clause contains a character mention as subject or object **and** a stative verb
    return True
Rule 3
if a clause is in progressive tense
    return False

Figure 6
Examples of manually designed rules.

extractive summaries produced by three people. Section 6 discusses the evaluation procedures in detail and reports the results.

5.2 Experiments with Manually Designed Rules

The first classification procedure applies manually designed rules to a clause-level representation of the original stories to produce descriptive summaries. The rules are designed using the same features as those used for machine learning and described in Section 4.3 and in Appendix A.

The first author created two sets of rules to guide the sentence classification process: one for the coarse-grained and another for the fine-grained representation. The rules operate at clause level. If a clause is deemed summary-worthy, the complete parent sentence is included in the summary. Figure 6 displays a few examples of rules for the fine-grained data set (a clause is considered to be summary-worthy if a rule returns *True*). The first rule attempts to select clauses that talk about one of the main characters and contain temporal expressions of type *enactment*. The rationale for this rule is that such clauses are likely to describe habitual activities of protagonists (e.g., *He always smoked*.) The second rule follows the same rationale but the stativity of the situation is signaled by the main stative verb. The third rule rejects clauses in progressive tense because such clauses are unlikely to contain background information.

The set of rules for the fine-grained representation has a tree-like structure. It processes the features of a clause and outputs a binary prediction. The rules for the coarse-grained representation function differently. Each clause is assigned a score based on the values of its features. The system then selects 6% of sentences that contain clauses with the highest scores. The scores attributed to the particular feature values were assigned and fine-tuned manually using linguistic knowledge described in Section 4.3. The reasons why the procedures for the two data sets differ are as follows. Assigning and fine-tuning the scores is a more flexible process and it is easier to perform manually. Ideally, we would apply score-based rules to both representations, but assigning and fine-tuning the scores manually for the fine-grained data set is excessively labor-intensive: there are too many features with too many values. For instance, one may want to reward clauses in simple past or present tenses, reflecting the fact that such clauses are more likely to be descriptive than those in perfect or progressive tenses. This information is expressed in the coarse-grained data set using one binary feature *simple_past_present* and fine-tuning the score is trivial. On the other hand, the same

– reject clauses without character mentions
– reject clauses in future tenses or in imperative mood
– reject clauses where the character mention is indefinite (e.g., *a man*)
– accept clauses containing an early character mention or a character mention modified by an apposition
– if a clause has not been rejected or accepted thus far, accept it if it is a background description

Figure 7
High-level overview of the rules for the fine-grained data set.

information in the fine-grained data set is distributed over three features with a total of seven values: *is_perf (yes, no)*, *is_progressive (yes, no)*, and *tense (past, present. future)*. Distributing the "reward" among three independent features is far less obvious.

The rules in both data sets, as well as the set of weights used for the coarse-grained representation, were selected and fine-tuned empirically using the training portion of the corpus as a guide. Once the parameters had been adjusted, the system produced two sets of summaries for the test portion of the corpus (one for each representation).

The detailed algorithms for both data sets are too long for inclusion in this article. Figures 7 and 8 show the rationale for the algorithms. The interested reader is referred to Kazantseva (2006) for pseudo-code.

5.3 Experiments with Machine Learning

As an alternative to rule construction, in the second set of experiments we performed decision tree induction with C5.0 (Quinlan 1992) to select salient descriptive sentences. C5.0 was our choice mainly because of the readability of its output.

The training and test data sets exhibited an almost 1:17 class imbalance (i.e., only 6% of all annotated clauses belonged to the positive class). Because the corpus was rather small, we applied a number of techniques to correct class imbalance in the training data set. These techniques included classification costs, undersampling (randomly removing instances of the majority class), oversampling (randomly duplicating instances of the minority class), and synthetic example generation (Chawlar et al. 2002). Using tenfold cross-validation on the training data set and original annotations by the first author,

– penalize clauses from sentences that are not assertive

– reward the following types of clauses:
  – clauses containing the first mention of a character or a mention occurring in the first 20% of sentences
  – clauses where the main verb is stative
  – clauses containing a temporal expression with long duration
  – clauses in simple past or present tense
  – clauses with a modal verb
  – clauses in active voice

Figure 8
High-level overview of the rules for the coarse-grained data set.

we selected the best class-imbalance correction techniques for each representation and also fine-tuned the learning parameters available in C5.0. These experiments brought the best results when using classification costs for the coarse-grained data set and undersampling for the fine-grained data set.

In order to see what features were most informative in each data set, we conducted a small experiment. We removed one feature at a time from the training set and used the decrease in F-score as a measure of informativeness. The experiment showed that in the coarse-grained data set the following features were the most informative: the presence of a character in a clause, the difference between the index of the current sentence and the sentence where the character was first mentioned, syntactic function of a character mention, index of the sentence, and tense. In the fine-grained data set the findings are similar: the index of the sentence, whether a character mention is a subject, the presence of a character mention in the clause, and whether the character mention is a pronoun are more important than the other features.

After selecting the best parameters on the training data set using tenfold cross-validation, the system produced two sets of summaries for the test data set.

6. Evaluation

6.1 Overview

We are not aware of any agreed-upon metrics for evaluating summaries of short fiction. In fact, it is not wholly clear what makes one summary better than another even for manual ones. That is why we evaluate our summaries using a variety of metrics and baselines, hoping to obtain a stereoscopic view of their quality.

The first evaluation experiment aims to measure the informativeness and the usefulness of the summaries. It is designed so as to compare the machine-made summaries with a random baseline and also with a "ceiling"— manual abstracts (henceforth **model summaries**). To achieve this, we engaged 15 evaluators to read the summaries and the stories of the test set and to answer two types of questions about them: factual (e.g., *list main characters of the story*) and subjective (e.g., *rank readability of the summary*). Such experimental design allowed us to evaluate extrinsically the informativeness of the summaries and intrinsically their usefulness. Both types of questions were asked first after reading the summary alone and then after reading the complete story. The summaries are an anonymous mix of random, machine-made, and model ones (i.e., the evaluators did not know whether the summaries were produced by programs or by people). Section 6.2 describes the experiment in detail.

The second round of evaluation aimed to evaluate the summaries by measuring sentence co-selection with the manually created extracts. It was designed to allow the comparison of machine-made summaries with two naive baselines and with two state-of-the-art generic summarizers (baseline summarizers). Section 6.3 contains the description of this experiment.

The third evaluation experiment compared the machine-made and the baseline summaries with the manually created abstracts using ROUGE (Lin 2004)—a package for automatically evaluating summaries. This experiment is described in Section 6.4.

6.2 Evaluating Informativeness and Usefulness of the Summaries

We define the objectives of this experiment as measuring the informativeness, the usefulness and, to some extent, the linguistic quality of the summaries that our system

produces. The informativeness is measured indirectly—by asking people factual questions about the story. The linguistic quality and the usefulness are evaluated intrinsically—by asking people to rank specific characteristics of the summaries. Fifteen unbiased evaluators answer both types of questions twice, first after reading the summary alone and then again after reading the complete story—repeating the procedure for all 20 test stories. Asking the questions after reading the summary alone measures the informativeness and the usefulness of the summaries in a realistic situation: To an evaluator, the summary is the only source of information about the original story. Repeating the procedure after reading the complete story evaluates the summaries in a situation where the evaluator has the complete information about the source. Each evaluator works with a mix of machine-made, random, and model summaries with six or seven summaries of each kind. This allows comparing the performance of our summarizer with a baseline and a ceiling.

Our summarizer produces four different flavors of summaries.[10] The labor intensity of the process prohibits asking the subjects to evaluate all four summary types. That is also why it is not possible to use more than one baseline or the summaries created by the baseline systems.[11] Restricted to evaluating only one type of the machine-made summaries, we opt for the coarse-grained rule-based ones, mainly because the coarse-grained representation and the rules make it easier to trace why the system selects specific sentences.

Conditions to be tested. We evaluate three factual and four subjective characteristics of the summaries.

Factual

How well the reader can name

- *the main characters*
- *the location*
- *the time*

when the summary is the only source of information about the story.

Subjective

- *How readable the summaries are.*
- *How much irrelevant information they contain.*
- *How complete they are.*
- *How useful they are for deciding whether to read the complete story.*

Evaluating these facets of the summaries reveals whether we achieve the objective of producing informative summaries. The focus of the system was not on readability. Still, we evaluate how readable the summaries are, because severe lack of coherence may prevent people from correctly interpreting the available information. We have provided

10 The options are as follows: either the coarse-grained or the fine-grained representation, and selecting sentences using either rules or machine learning.

11 Each additional summary type would have required approximately 67 extra man-hours.

no definitions or criteria to the subjects apart from the questions shown in Tables 4 and 5.

Baselines. We compare the machine-made summaries with a baseline and a ceiling. The baseline consists of randomly selected sentences. Both the machine-made and the random summaries contain the same number of sentences. The ceiling consists of the summaries written by two human subjects. The summary writers were instructed to write 20 summaries of short stories in a way that does not reveal all of the plot. They received one example summary and were allowed to reuse sentences from the stories, to employ metaphor and any other literary devices they found useful.

Metrics. The evaluators answered the factual questions about the main characters and the location of the story in their own words. The first author rated the answers on the scale of -1 to 3. A score of 3 means that the answer is complete and correct, 2 = slightly incomplete, 1 = very incomplete, 0 = the subject cannot find the answer in the text, and -1 = the answer is incorrect. The question asking to identify the time frame of the story is a multiple-choice one: select the century when the story takes place. The answers to this question are rated on a binary scale (1 if the answer is correct, 0 if it is not or if the subject cannot infer time from the text). We calculate the mean answers for each question and compare them across summary types using the Kruskal–Wallis test and the Mann–Whitney test (also known as the Wilcoxon Rank–Sum test). The tests are appropriate when the response variable is ordinal and the dependent variable is categorical. Both tests are based on assigning ranks to the available data points.

The Kruskal–Wallis test is a nonparametric test used to determine whether several samples come from the same population. It is based on calculating the K statistic which follows χ^2 distribution for sample sizes of five or larger. Given i samples containing t_i data points each with R_i being the sum of ranks of all data points in sample t_i, K is calculated as follows (Leach 1979, page 150):

$$K = \frac{12}{n(n+1)} \sum t_i \left(\frac{R_i}{t_i} - \frac{n+1}{2}\right)^2 \quad (1)$$

In order to make pairwise comparisons between samples and to establish the locus of the difference, we rely on the Mann–Whitney test. The test is based on calculating the S statistic. For large sample sizes the distribution of S can be approximated using the normal distribution.

$$S = 2R - t_2(n+1) \quad (2)$$

where t_2 is the size of the smaller sample, n is the size of both samples together, and R is the sum of ranks in the smaller sample. We use the Kruskal–Wallis test with 0.01 confidence level. In order to avoid increasing the chance of Type I error when performing pairwise comparisons, we set per-comparison confidence level for the Mann–Whitney test at $\beta = \gamma/c$ where γ is the desired per-experiment confidence level and c is the number of comparisons (Leach 1979, page 161). In our case $\beta = 0.0033$.

All subjective questions are multiple-choice questions. An evaluator has to select a score of 1 to 6, with 1 indicating a strong negative property and 6 indicating a strong positive property. We opt for a scale with an even number of available values so as to avoid the evaluators' giving excessive preference to the middle rank. We measure the mean ranks for each question and compare them across summary types using the Kruskal–Wallis and Mann–Whitney tests. The inter-annotator agreement is computed

using Krippendorff's α (Krippendorff 2004, pp. 221–236) (henceforth α). α measures disagreement between annotators corrected for chance disagreement.

$$\alpha = 1 - \frac{Disagreement_{observed}}{Disagreement_{expected}} = 1 - \frac{Average\_metric\_\delta^2\_within\_all\_categories}{Average\_metric\_\delta^2\_within\_all\_items} \quad (3)$$

Unlike other coefficients of inter-coder agreement, α allows taking into account the magnitude of disagreement by specifying a distance metric δ^2. This property is crucial in our case: a situation when raters disagree whether to give a summary a rank of 1 or 6 should be penalized more heavily than a situation when they do not agree between the ranks of 5 and 6. When computing α, we use the distance metric suggested by Krippendorff for ordinal data (Krippendorff 2004, page 223):

$$ordinal\_\delta^2 = (\frac{n_c}{2} + \sum_{g>c}^{g<k} n_g + \frac{n_k}{2})^2 \quad (4)$$

where c and k, $c < k$, are the two ranks.

For all questions, the computation of α is based on the following parameters: N = 300, n = 15, and c = 6, where N is the total number of items (i.e., summary–story pairs ranked), n is the number of raters, and c is the number of available categories.

Subjects. The participants for the experiment were recruited by the means of advertising at the Department of Linguistics at the University of Ottawa. Most of them are third- and fourth-year undergraduate students of linguistics. The only requirement for participation was to be a native speaker of English. We hired two people to create model summaries for the 20 stories of the test set. The summary writers worked approximately 15–20 hours each. Fifteen people were hired to evaluate the summaries (i.e., to read the summary–story pairs and answer the questions). The task of evaluating a summary required approximately 12–15 hours of labor per person. All participants were paid. The instructions for summary writers are available at `www.site.uottawa.ca/~ankazant/instructions-writers.zip`. The instructions for evaluators can be found at `www.site.uottawa.ca/~ankazant/instructions_eval.zip`.

Material. Each evaluator received 20 summary–story pairs. Because some questions sought to measure the informativeness of the summary, every evaluator worked on 20 distinct stories of the test set and no one worked with the same story more than once. The summaries were a randomly selected mix of random, machine-made, and model summaries.

Procedure. The experiment was conducted remotely. The summary writers received the test set of stories and the instructions and had seven working days to submit their abstracts. A week later, we sent randomly generated packages of summary–story pairs to the evaluators. The packages contained between six and seven summaries of each kind (random, machine-made, and model). Each evaluator worked with exactly one summary for each story, reading a total of 20 pairs. Every summary was evaluated by five subjects. The evaluators had seven working days to complete the task.

Results. Informativeness. Table 4 shows the results of comparing the mean answers between the machine-made, the baseline, and the model summaries using the Kruskal–Wallis and Mann–Whitney tests. The column *Groups* shows homogeneous groups, identified using the Mann–Whitney test with 99.67% confidence (recall that per-comparison confidence level $\beta = 0.0033$). The groups are denoted using distinct literals (e.g., *A*, *B*, *C*).

Table 4
Answers to factual questions.

Summary type	*After reading the summaries only*			*After reading the stories*		
	Mean rank	Groups	Std. Dev	Mean rank	Groups	Std. Dev
Question: Name three main characters in the order of importance.						
Model	2.24	A	0.73	2.73	A	0.49
Machine	2.21	A	0.69	2.71	A	0.56
Random	1.42	B	1.04	2.67	A	0.62
Question: Name the location of the story.						
Model	2.1	A	1.25	2.62	A	0.93
Machine	1.39	B	1.33	2.79	A	0.62
Random	0.71	C	1.18	2.43	A	0.98
Question: Select the century when the story takes place.						
Model	0.5	A	0.5	0.69	A	0.46
Machine	0.29	B	0.46	0.76	A	0.43
Random	0.19	B	0.39	0.7	A	0.54

The differences between the machine-made summaries and the random ones are significant for the questions about characters and the location of the story. This shows that in these respects the machine-made summaries are—rather predictably—consistently more informative than the random ones. The difference between the machine-made and the random summaries is not statistically significant for the question asking to name the time of the story. Keeping in mind how rare absolute temporal anchors are in short stories, this is not surprising. The manual summaries, however, are ranked higher with statistical significance. This may suggest that the machine-made summaries are not as coherent as the model ones, which prevents the reader from finding implicit cues about timeframe available in the summaries.

The differences between the machine-made and the model summaries are significant for the questions about the time and the place of the story, but not for the questions about the main characters. This suggests that the machine-made summaries are almost as informative as the model ones when it comes to informing the reader whom the story is about. They cannot, however, give the reader as good an idea about the time and the place of the story as the model summaries can.

All summary types are less informative than the complete story; that is, the differences between answers obtained after reading the summary alone and after reading the complete story are significant in all cases.

Usefulness and linguistic quality. Table 5 shows mean ranks for the three summary types, along with the homogeneous groups identified using the Mann–Whitney test, with 99.67% confidence. The request to rank readability was made only once—after reading the summary; the request to evaluate the completeness was made only after reading the complete story. (The corresponding cells in Table 5 are empty.)

The results reveal that the evaluators consistently rank the model summaries as best, the machine-made summaries as second-best, and the random ones as worst. The differences between summary types are significant in all cases.

The readability of the machine-made summaries is ranked as slightly better than average (3.28 on the scale of 1 to 6). For all other questions (relevance, completeness, and

Table 5
Subjective rankings.

Summary type	*After reading the summaries only*			*After reading the stories*		
	Mean rank	Groups	Alpha	Mean rank	Groups	Alpha
Question: How readable do you find the summary? (Is it coherent and easy to read, or confusing and does not make sense?) (scale: 1 to 6)						
Model	5.44	A	0.74			
Machine	3.28	B				
Random	1.89	C				
Question: How much irrelevant information does the summary contain? (useless, confusing information, fragments that do not make sense) (scale: 1 to 6)						
Model	5.10	A	0.61	5.24	A	0.62
Machine	2.83	B		2.82	B	
Random	1.93	C		1.85	C	
Having read the story, do you find that a lot of important information is missing? Rate how complete you find the summary. (scale: 1 to 6)						
Model				5.18	A	0.69
Machine				2.81	B	
Random				1.65	C	
Imagine that this summary should help you decide whether you would like to read the complete story. How helpful was the summary for this purpose? (scale: 1 to 6)						
Model	5.22	A	0.60	5.11	A	0.63
Machine	2.81	B		2.81	B	
Random	1.88	C		1.65	C	

usefulness), the machine-made summaries are ranked as slightly worse than average (around 2.81). This shows that even though the summaries are somewhat useful and consistently outperform the random baseline, they fall short of the quality of the manual abstracts. This is hardly surprising given the inherent difficulty of summarizing fiction and the exploratory nature of this work. It is worth remarking that even the model summaries do not appear to be perfect: The evaluators ranked them around 5.20, even though they had significantly worse summaries to compare against. This may suggest that the task is not easy even for people, let alone for a computer program.

The column labelled *Alpha* in Table 5 shows the results of measuring the extent to which the evaluators agree when answering the subjective questions.[12] The agreement is measured using Krippendorff's α. The results show substantial agreement but fall short of the reliability cut-off point of 0.8 suggested by Krippendorff. The failure to reach such high agreement is hardly surprising: the task of ranking the quality of the summaries is highly subjective. Instead of asking the subjects to bin items into a predefined number of categories, the task calls for discretizing a concept which is

12 We have not measured the agreement for factual questions because those questions were answered in the evaluators' own words and the answers were scored by the first author. To give the reader an idea of variability of the answers, Table 4 reports standard deviation from the mean.

continuous in nature: the quality of a summary. That is why we interpret the level of agreement as sufficient for the purpose of evaluating the quality of the summaries.

6.3 Comparing the Machine-Made Summaries and the Manually Created Extracts

Measuring sentence co-selection between extractive summaries created by humans and those created by automatic summarizers has a long tradition in the text summarization community (Lin and Hovy 2000; Marcu 2000), but this family of measures has a number of well-known shortcomings. As many have remarked on previous occasions (Mani 2001; Radev et al. 2003), co-selection measures do not provide a complete assessment of the quality of a summary. First of all, when a summary in question contains sentences that do not appear in any of the model extracts, one may not be sure that those sentences are uninformative or inappropriate for inclusion in a summary. In addition, documents have internal discourse structure and sentences are often inter-dependent. Therefore, even if a summary contains sentences found in one or more reference summaries, it does not always mean that it is advisable to include those sentences in the summary in question.

Sentence co-selection does not explicitly measure the quality of a summary. It does, however, measure a quality that is objective and easy to pin down: how many sentences that humans judge summary-worthy are included in the machine-made summary. Such a metric is a useful complement to the results reported in Section 6.2. It has the advantage of being easy to interpret and comprehend. It also has a long tradition of usage which allows us to compare our summarizer—on a familiar scale—with other summarization systems. That is why we chose co-selection as the basis for comparing the summaries that our system produces with manually created extracts.[13]

Overview. The experiment involves six annotators divided into two groups of three. Each annotator is asked to read 10 short stories and to select 6% of sentences that, in their opinion, constitute a good indicative summary. In this manner three people annotate each story of the test set for summary-worthy sentences. We used their annotations as a gold standard and compared the machine-made summaries against them. In addition, we used the same gold standard and metrics to evaluate the quality of two baseline summaries and of two summaries produced by state-of-the art summarization systems.

Conditions to be tested. The purpose of the experiment is defined as measuring how many sentences found in our system's summaries and the baseline summaries occur in the extractive summaries created by the human annotators. We are also interested in finding out whether our summarizer outperforms the trivial baseline algorithms and the existing state-of-the-art summarizers fine-tuned to summarizing newswire.

Baselines. In order to evaluate our summarizer comparatively, we defined two naive baselines and a ceiling. Intuitively, when a person wishes to decide whether to read a book, she opens it and flips through several pages at the beginning. Imitating this process, we computed a simple lead baseline consisting of the first 6% of the sentences in a story. The second baseline consists of 6% of sentences of the story selected at random. The ceiling consists of all sentences deemed summary-worthy by one of the human annotators.

13 We decided against using deeper approaches, such as the Pyramid method (Nenkova and Passonneau 2004), factoids (van Halteren and Teufel 2003), and relative utility (Radev and Tam 2003). The reason is practical: These approaches have an unfortunate disadvantage of being considerably more labor-intensive than the measures based on sentence co-selection.

It is also necessary to see whether our genre-specific approach shows any improvements over the existing generic state-of-the-art systems put to work on fiction. To this end, we compared our summarizer with two systems that were top performers in the Document Understanding Conference (henceforth DUC) 2007, the annual "competition" for automatic summarizers. In DUC competitions the summarization systems are evaluated on a variety of metrics: manually assigned scores (ranking readability, grammaticality, non-redundancy, referential clarity, focus, and coherence), the pyramid method (Nenkova and Passonneau 2004), and ROUGE scores (Lin 2004). There is no unified ranking of the systems' performance, and selecting the best summarizer is not straightforward. We chose two systems among the top performers in DUC 2007—GISTexter (Harabagiu, Hickl, and Lacatusu 2007) and CLASSY (Schlesinger, O'Leary, and Conroy 2008; Conroy, Schlesinger, and O'Leary 2007). GISTexter appears to be the best summarizer according to the scores assigned by the human judges. Apart from baselines, it is consistently ranked as the best or the second-best system on most characteristics evaluated by the judges (the only exception is non-redundancy where GISTexter is ranked eighth). CLASSY, on the other hand, is one of the four top systems according to ROUGE scores. The scores it received from the human judges are also quite good.

The main task in DUC 2007 called for creating 250-word summaries from a collection of newswire articles on a specific topic. Each input collection was accompanied by a topic statement that briefly explained what the summaries should cover. Therefore, both CLASSY and GISTexter are geared towards multi-document query-based summarization of newswire—a task dramatically different from that of summarizing short fiction.[14] No adjustments were made to either system to make them more suitable for summarizing stories. Therefore, the comparison is not wholly fair, but—in the absence of systems similar to ours—it was the only possibility to compare our summarizer with the state-of-the-art in the community and to verify whether genre-specific methods are useful in summarizing fiction.

Evaluation metrics. By combining summaries created by the annotators in several ways we create three distinct gold-standard summaries. The **majority** gold-standard summary contains all sentences selected by at least two judges. It is best suited to give an overall picture of how similar the machine-made summaries are to the man-made ones. The **union** gold standard is obtained by considering all sentences that are judged summary-worthy by at least one judge. Union summaries provide a more relaxed measurement. Precision computed on the basis of the union gold standard gives an idea of how many irrelevant sentences a given summary contains (sentences not selected by any of the three judges are more likely to prove irrelevant). The **intersection** summaries are obtained by combining sentences that all three judges deemed to be important. Recall measured on the basis of the intersection gold standard says how many of the most important sentences are included in summaries produced by the system (sentences selected by all three judges are likely to be the most important ones). All summaries are compared against each gold-standard summary using precision (P), recall (R), and equally-weighted F-score (F).

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

14 GISTexter in particular relies on extensive syntactic and semantic query decomposition and, thus, is at a disadvantage when no informative query is provided.

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$F = \frac{2TP}{2TP + FP + FN} \tag{7}$$

TP denotes true positives, FP = false positives, TN = true negatives, and FN = false negatives.

The statistical significance of the differences between various types of summaries is established using the one-way Analysis Of Variance test (ANOVA) and the Tukey Honestly Significant Differences test (henceforth **Tukey HSD**).

ANOVA tests whether the differences between sample means are significant by comparing variance between samples with total variance within samples:

$$f = \frac{S_{between}^2/(p-1)}{S_{within}^2/(n-1)} \tag{8}$$

where p is the number of samples, n is the total number of observations, $S_{between}$ is the sum of squared deviations between sample means and the total mean, and S_{within} is the total sum of squared deviations within samples. The f statistic is distributed as F when the null hypothesis is true (i.e., the differences between sample means are not significant).

The power of ANOVA extends as far as verifying whether the differences between sample means are significant overall, but the test does not say anything about differences between particular pairs of sample means. Tukey HSD is a test that does just that. It measures q, the studentized range statistic:

$$q = \frac{M_l - M_s}{SE} \tag{9}$$

where M_l is the larger of the sample means, M_s is the smaller one, and SE is the standard error of the data in question. In accordance with our interpretation of the three types of gold standards, we use the most meaningful measurement for each standard: F-score for the majority, precision for the union, and recall for the intersection. We also use the same measurement to set the ceiling for each standard (i.e., by choosing the manually created extract that compares best on that scale).

Before combining the extracts created by the annotators into gold-standard summaries, we measure how well these people agree among themselves. We estimate the agreement using Scott's π (Scott 1955).[15] This coefficient measures the observed agreement between judges corrected for chance agreement.

$$\pi = \frac{Agreement_{observed} - Agreement_{expected}}{1 - Agreement_{expected}} \tag{10}$$

15 The coefficient is also known as Siegel and Castellan's κ (Siegel and Castellan 1988).

Table 6
Inter-annotator agreement on selecting summary-worthy sentences.

Statistic	Group 1	Group 2	Average
$\pi(4)$	0.52	0.34	0.43
$\pi(3)$	0.50	0.34	0.42

$$Agreement_{observed} = \frac{1}{ic(c-1)} \sum_{i \in I} \sum_{k \in K} n_{ik}(n_{ik} - 1) \tag{11}$$

$$Agreement_{expected} = \frac{1}{(ic)^2} \sum_{k \in K} n_k^2 \tag{12}$$

where i is the number of items to be classified in set I, k is the number of available categories in set K, c is the number of coders, n_{ik} is the number of coders who assign item i to category k, and n_k is the total number of items assigned to category k by all annotators (Artstein and Poesio 2008, pp. 562–563).

Subjects. Six human subjects participated in annotating the test set of stories for the presence of summary-worthy sentences. These people are colleagues and acquaintances of the first author. At the time of the experiment none of them was familiar with the design of the system. Four annotators are native speakers of English and the remaining two have a very good command of the language.

Materials. The material for the experiment consisted of the 20 stories of the test set. Three annotators created extractive summaries for each story. In addition, there were eight distinct automatically produced summaries per story: four summaries produced by our system, two baseline summaries, and two summaries created by the baseline systems from DUC.

Procedure. The experiment was conducted by e-mail. The annotators received the stories and had two weeks to annotate them. The participants reported having taken 10–20 hours to complete the task.

Results. Agreement. Table 6 reports the agreement between the judges within each group and with the first author of this article. The agreement with the first author is reported because she created the initial training and test data for experiments. The numbers 3 and 4 state whether the statistic is computed only for three subjects participating in the evaluation or for four subjects (including the first author). As can be seen from Table 6, the agreement statistics are computed for each group separately. This is because the sets of stories that they annotated are disjoint. The "Average" column shows an average of these figures, to give a better overall idea.

The agreement values in Table 6 are rather low. They fall well below the 0.8 cut-off point specified by Krippendorff (2004). On a less demanding scale, Landis and Koch (1977) interpret values in the range of 0.21–0.4 as fair agreement and in the range of 0.41–0.6 as moderate agreement.[16] Weak agreement is not surprising: Many researchers

16 Krippendorff's suggestion refers to α, rather than Scott's π, and Landis and Koch's scale was created for Cohen's κ (Cohen 1960). In our setting, however, the values of κ, π and α are almost the same.

report that people do not agree well on what sentences constitute a good summary of a document (Rath, Resnick, and Savage 1961; Salton et al. 1997; Lin and Hovy 2003). In most cases the agreement corresponding to π of 0.42 would not be sufficient for creating a resource, but we interpret this level of agreement as acceptable for evaluating a single facet of the summaries that are also evaluated in other ways.

Co-selection. Tables 7–9 show the results of comparing eight different versions of the computer-made summaries against the gold-standard summaries produced by people. In each table, the entry *HUMAN* corresponds to the summaries created by the annotator who achieves the highest scores for the corresponding standard. The "Groups (*metric*)" column reports homogeneous groups identified using Tukey HSD with 95% confidence for the specified metric.

Our system outperforms both baseline algorithms and the baseline summarizers, but it always falls short of the performance of the best human summary. The improvement margins between the random and the baseline systems' summaries and those produced by our system are rather wide. The weaker performance of the baseline summarizers strongly suggests the need for genre-specific methods when summarizing short fiction.

The differences between the lead summaries and the system-made ones are also statistically significant, yet they are much narrower. We interpret this as an indication that the lead baseline is more demanding than the random one when creating indicative summaries of short fiction.

Table 7
Sentence co-selection between computer- and human-made summaries. Majority gold standard.

Data set	Precision	Recall	F	Groups (F)
HUMAN	*64.95*	*84.18*	*72.41*	*A*
Rules, fine-grained	39.20	53.35	44.47	B
Machine-learning, fine-grained	36.36	49.44	41.26	B
Rules, coarse-grained	35.42	44.39	38.94	B
Machine learning, coarse-grained	35.31	41.81	36.90	B
LEAD	*25.50*	*31.18*	*27.57*	*C*
CLASSY	7.14	9.08	7.70	D
GISTexter	9.67	6.30	6.91	D
RANDOM	*4.10*	*5.40*	*4.57*	*D*

Table 8
Sentence co-selection between computer- and human-made summaries. Union gold standard.

Data set	Precision	Recall	F	Groups (P)
HUMAN	*1*	*55.61*	*71.05*	*A*
Rules, fine-grained	56.10	32.11	40.56	BC
Rules, coarse-grained	53.98	29.77	38.04	BC
Machine-learning, fine-grained	52.39	30.66	38.35	C
Machine learning, coarse-grained	49.62	25.06	32.78	C
LEAD	*36.76*	*18.53*	*24.50*	*DEF*
GISTexter	22.70	6.18	9.28	EFG
CLASSY	19.74	10.23	13.05	FG
RANDOM	*12.41*	*6.40*	*8.40*	*G*

Table 9
Sentence co-selection between computer- and human-made summaries. Intersection gold standard.

Data set	Precision	Recall	F	Groups (R)
HUMAN	*31.32*	*1.00*	*45.21*	*A*
Rules, fine-grained	23.11	79.25	34.02	AB
Rules, coarse-grained	21.70	65.42	30.55	BC
Machine learning, coarse-grained	19.66	57.29	26.80	BC
Machine-learning, fine-grained	16.41	56.19	24.00	BC
LEAD	*12.37*	*38.83*	*17.84*	*DE*
CLASSY	3.58	8.04	4.89	F
GISTexter	3.20	6.06	3.83	F
RANDOM	*0.79*	*1.67*	*1.06*	*F*

The results also suggest that automatically produced summaries bear some resemblance to manual ones. There is no straightforward way to interpret these results as good or bad in the context of other summarization systems. Firstly, the task is new and no comparable results exist. Secondly, even though sentence co-selection metrics have been widely used for evaluating summaries of other genres, different compression rates, different gold standards, and availability of naturally occurring competitive baselines (e.g., lead baseline in newswire summarization) make fair comparison difficult. For example, Marcu (2000, page 214) reports achieving F-score of 76.04 when creating summaries of newswire articles at 10% of their original length. The lead baseline achieves F-score of 71.89. When summarizing dialogues, Zechner (2002, page 479) reports weighted accuracy of 0.614 compared to the lead baseline's performance of 0.438 (the numbers are averages over five different summary sizes of 5%, 10%, 15%, 20%, and 25%). In this context we interpret the results in Tables 7–9 as suggesting that our genre-specific system outperforms the naive baselines and two generic summarizers.

6.4 Evaluating Summaries using Lexical Overlap

ROUGE (Lin 2004) is a package for automatically evaluating summaries. Given one or more gold-standard summaries (usually written by people), ROUGE offers several metrics for evaluating the summary in question. The metrics reward lexical overlap between the model summaries and the candidate one. Depending on the metric, the lexical units taken into consideration are *n*-grams, word sequences, and word pairs.

Since 2004, ROUGE scores have been among the measures used for evaluating automatic summarizers at DUC. Following this tradition, we ran ROUGE to evaluate our summaries and to compare them to the baselines (including CLASSY and GISTexter).

Conditions to be tested. The objective of the experiment was to establish how much lexical overlap exists between the machine-made and the model summaries. We achieved this by computing ROUGE-2 and ROUGE-SU4 scores.[17]

Baselines and material. We evaluated eight types of summaries: four types created by our summarizer, the lead and the random baselines, and the summaries created by

17 ROUGE-2 and ROUGE-SU4 are two measures that were used at DUC 2007.

GISTexter and CLASSY. In addition, we included a ceiling by computing ROUGE scores for the model summaries.

Metrics. ROUGE-2 score measures the bigram recall between the reference summary and the candidate one. It is computed according to the following formula:

$$ROUGE\text{-}2 = \frac{\sum_{s \in S} \sum_{b \in s} Count_{match}(b)}{\sum_{s \in S} \sum_{b \in s} Count(b)} \quad (13)$$

where S is the set of reference summaries, b is a bigram in the reference summary s, $Count_{match}(b)$ is the number of bigrams that both summaries share, and $Count(b)$ is the total number of bigrams in the reference summary s.

ROUGE-S measures the similarity of a pair of summaries based on how many **skip-bigrams** they have in common. A skip-bigram is any pair of words in a sentence, allowing for arbitrary gaps.

$$ROUGE\text{-}S = \frac{SKIP2(X,Y)}{C(m,2)} \quad (14)$$

where X is the reference summary of length m, Y is the candidate summary, $SKIP2(X,Y)$ is the number of skip-bigram matches between X and Y, and C is the combination function.

ROUGE-SU4 is an extension of ROUGE-S that also rewards matching unigrams. The maximum gap allowed by skip-bigrams is 4 (hence SU4).

In order to compare the automatically produced summaries with those created by humans we implemented the following leave-one-out procedure. At first, we computed ROUGE scores by comparing all automatically produced summaries (i.e., those created by our system and the baseline ones) and one of the model summaries against the second available model summary. Next, the procedure was repeated but the model summaries were switched. The significance of the differences was tested using ANOVA and Tukey HSD for 95% confidence level. When calculating ANOVA and Tukey HSD, we used the scores obtained from both runs.

Results. Tables 10 and 11 show ROUGE-2 and ROUGE-SU4 scores for all automatically produced and model summaries. The results are inconclusive.

When using ROUGE-2 as a guide, the only summaries consistently different from the rest with 95% confidence are the randomly generated ones. The scores of all other summaries are too close to reject the hypothesis that the differences are due to chance. This is the case even with the differences between the model and the automatically produced summaries. A possible interpretation could be that all summaries are of very high quality that is indistinguishable from that of the model summaries. This hypothesis, however, can be easily dismissed: The results reported in Sections 6.2 and 6.3 clearly show that the quality of the summaries produced by our system is well below the ceiling.

The situation is similar with ROUGE-SU4 scores, if not so dramatic. There are three distinct groups of summaries. Group *A* includes the rule-based fine-grained summaries and those produced by CLASSY. The second group includes the lead baseline, three types of summaries created by our summarizer, the model summaries, and those created by GISTexter. The last group contains the random and the lead baselines. Even though ROUGE-SU4 measurement seems to have more discriminative power, it is at least puzzling that it cannot distinguish between the model and the automatically

Table 10
ROUGE-2 recall scores.

System	ROUGE-2	Groups
HUMAN-1	*0.0874*	*A*
HUMAN-2	*0.0807*	*A*
Rules, fine-grained	0.0981	A
Machine learning, fine-grained	0.0905	A
GISTexter	0.0829	A
CLASSY	0.0826	A
Rules, coarse-grained	0.0816	A
Machine learning, coarse-grained	0.0808	A
LEAD	*0.0572*	*AB*
RANDOM	*0.038*	*B*

Table 11
ROUGE-SU4 recall scores.

System	ROUGE-2	Groups
Rules, fine-grained	0.1684	A
CLASSY	0.1654	A
GISTexter	0.1607	AB
Rules, coarse-grained	0.1564	AB
HUMAN-1	*0.1540*	*AB*
Machine learning, coarse-grained	0.1468	AB
HUMAN-2	*0.1426*	*AB*
Machine learning, fine-grained	0.1584	AB
LEAD	*0.127*	*BC*
RANDOM	*0.0956*	*C*

produced summaries. In particular, placing the rule-based coarse-grained summaries and the model ones in the same group directly contradicts the results reported in Section 6.2—that people find the model summaries far superior to this particular type of summary produced by our summarizer.

We interpret these results as suggesting that the ROUGE-2 and ROUGE-SU4 scores are not well suited for evaluating indicative summaries of short stories. An explanation could be that when people summarize fiction—rather than newswire or scientific papers—they seem to use fewer sentences and clauses verbatim and, by and large, introduce more generalization and abstraction. (We have made this informal observation when processing the model summaries used in this experiment.) This results in little lexical overlap with the source text and hence with extractive summaries of any flavor. This hypothesis, however, is only preliminary and requires further investigation.[18]

18 It is possible that significantly increasing the number of model summaries would alleviate the problem. Unfortunately, obtaining so many model summaries was prohibitively expensive in our case. To move in this direction, we ran ROUGE without jackknifing to enable the use of two model summaries for comparison. The results were similar to those reported in Tables 10 and 11: Only the random summaries are consistently significantly worse than the rest.

7. Conclusions

We presented an approach to summarizing literary short stories. The text summarization community has not yet seriously explored this genre, except for early seminal work on story understanding. In contrast with the story-understanding systems proposed in the 1970s and 1980s, our system does not require labor-intensive semantic resources—knowledge-bases and schemas—and it works on real-life stories, namely, short fiction.

The summaries that the system produces, limited in scope, are intended to help readers form adequate expectations about the original story. We have demonstrated that such summaries can be produced without deep semantic resources, only relying on syntax and the information about important entities in the story. According to the judges who evaluated the summaries, our summaries are somewhat useful for their original purpose, even if their quality falls far short of the quality of manual abstracts. Our summaries appear better than the naive baselines and than two state-of-the-art summarizers fine-tuned for working with newswire.

In the course of this work we have made a number of observations about automatic summarization of short stories. First of all, we confirmed informally that characters tend to be a central element of short fiction. Character mentions provide a wealth of information that can be leveraged in automatic summarization. This finding was also reflected in the approach proposed by Lehnert (1982). In addition, it appears that position in text is important, as can be seen from the analysis of the usefulness of features in Section 5.3. Besides, relatively high performance of the lead baselines also suggests that position in text is a good indicator of salience, even though it plays a lesser role than in more structured documents.

We view this work as a small step towards creating tools for searching, summarizing, and otherwise processing fiction available electronically. The current system accomplishes with some success a limited task of producing indicative summaries or short stories, but much more work is needed to create high-quality flexible summaries of literary works suitable for more than one purpose. Perhaps the most obvious extension to the current system would be summarizing the plot of short stories. Although this is not useful given our original criterion (forming adequate expectations about the story, without "spoilers"), the ability to handle plot would allow the creation of different types of summaries. We also hope to explore the possibility of establishing structure within stories: Knowing that certain portions of a story lay out the setting while others describe events or the culmination would be a significant step towards better summarization.

Evaluation of summaries of literary work is yet another dimension of the task that needs to be considered. We have concentrated thus far on summary production rather than on establishing the criteria that define the quality of the summary. Evaluation of summaries remains an issue even where well-structured factual documents are concerned. In fiction, it is far less clear what contributes towards the quality of the summary: The facts, for instance, are likely to be less important than in scientific papers or news items. Other candidate qualities may include closeness to the language or the tone of the original story, the information about the author, the time period, or ideology behind a certain work of fiction. This remains an open question, the answer to which may well lie outside the field of computational linguistics.

Appendix A: Features Used in the Coarse- and the Fine-Grained Clause Representations

The appendix lists features computed to represent a clause in the fine-grained data set (Table 12) and in the coarse-grained data set (Table 13). Prior to constructing feature vectors, the stories are parsed with the Connexor Machinese Parser. All syntactic information is computed on the basis of the parser output. The "Category" column shows whether a feature is character-related (*C*), location-related (*L*), aspect-related (*A*), or other (*O*). LCS refers to the database of Lexical Conceptual Structures (Dorr and Olsen 1997).

Table 12
Features representing a clause in the fine-grained data set.

Name	Category	Possible values	Description	Default value
char_if_ind_obj	C	*yes*, *no*	*yes* if the clause contains a mention of a character and its grammatical function is indirect object	*no*
char_if_obj	C	*yes*, *no*	*yes* if the clause contains a mention of a character and its grammatical function is direct object	*no*
char_if_subj	C	*yes*, *no*	*yes* if the clause contains a mention of a character and its grammatical function is subject	*no*
char_in_sent	C	*yes*, *no*	*yes* if the parent sentence contains a mention of a character	*no*
char_indef	C	*def*, *indef*	*def* if the clause contains a mention of a character and a) it is a proper name or b) it is modified by a definite determiner or a pronoun; *indef* if the mention is modified by an indefinite determiner	*n/a*
char_is_attr	C	*yes*, *no*	*yes* if the mention of a character is in the genitive case	*n/a*
char_mention	C	*yes*, *no*	*yes* if the clause contains a mention of a character	*no*
char_modified	C	*yes*, *no*	*yes* if the mention of a character is modified by a noun phrase	*n/a*
char_pronoun	C	*1st*, *3rd*	*1st* if the clause contains a pronominal mention of a character and it is in 1st person (e.g., *I*); *3rd* if the pronominal mention is in 3rd person (e.g., *he*)	*n/a*
nbr_after_first_mention	C	continuous	an integer that reflects the difference between the index of the current sentence and the sentence where the character is first mentioned (it is only defined for clauses containing mentions of characters)	*-1*
loc_in_prep	L	*yes*, *no*	*yes* if the clause contains a mention of a location and is embedded in a prepositional clause	*no*

Table 12
(*continued*)

Name	Category	Possible values	Description	Default value
loc_present	L	*yes*, *no*	*yes* if the clause contains a mention of a location	*no*
durative	A	*yes*, *no*	*yes* if the main verb of the clause is durative; this information is computed using LCS	*no*
dynamic	A	*yes*, *no*	*yes* if the main verb of the clause is dynamic; this information is computed using LCS	*no*
modal	A	*can*, *could*, *shall*, *should*, *would*, *must*, *may*, *might*, *dare*, *need*, *will*, *ought*, *canst*	a modal verb from the list, if it appears in the clause	*n/a*
neg	A	*yes*, *no*	*yes* if the main verb of the clause is negated	*no*
obj_def	A	*yes*, *no*	*no* if the direct object of the main verb is modified by an indefinite determiner; *yes* in all other cases where a direct object is present	*n/a*
obj_plur	A	*yes*, *no*	*yes* if the direct object of the verb is in plural; *no* in all other cases where a direct object is present	*n/a*
passive	A	*yes*, *no*	*yes* if the clause is realized in passive voice	*no*
perf	A	*yes*, *no*	*yes* if the clause is realized in a perfect tense	*no*
progr	A	*yes*, *no*	*yes* if the clause is realized in a progressive tense	*no*
telic	A	*yes*, *no*	*yes* if the main verb of the clause is telic; this information is computed using LCS	*no*
tense	A	*past*, *present*, *future*	the tense used in the clause	*n/a*
tmp_magn	A	*min*, *hour*, *day*, *week*, *month*, *year*, *year_plus*	the magnitude of the core temporal unit in the expression (defined for clauses containing temporal expressions and assigned using a set of manually designed templates): *min* if the core unit denotes a period of no more than a minute (e.g., *in a few seconds, that moment*); *hour* if it denotes a period of no more than an hour (e.g., *during those hours, at 10 am*); the values *day* through *year* are assigned analogously, and *year_plus* denotes periods longer than a year (e.g., *for decades*)	*n/a*
tmp_plur	A	*yes*, *no*	*yes* if the core temporal unit in the expression is in plural (e.g., during those years), *no* if it is singular (e.g., that day); defined for clauses containing temporal expressions	*n/a*

Table 12
(*continued*)

Name	Category	Possible values	Description	Default value
tmp_type	A	*location, duration, frequency, enactment, temporal_ manner*	the type of the expression (defined for clauses containing temporal expressions, and assigned using a set of manually designed templates): all values except *temporal manner* are assigned according to the classification of temporal expressions available in the linguistic literature (Harkness 1987), for example *today (location)*, *during those hours (duration)*, *every day (frequency)*, *never (enactment)*; *temporal_ manner* is a separate pseudo-category defined to include expressions such as *immediately, instantly* etc.	*n/a*
clause_type	O	*assertive, imperative, infinitive, subjunctive*	the form of the main verb in the clause as output by the parser: *imperative* for clauses realized in the imperative mood, *subjunctive* for those realized in subjunctive, *infinitive* for infinitival clauses (e.g., *He decided to go*), *assertive* otherwise	*assertive*
nbr_of_sent	O	continuous	the index of the parent sentence in text	*-1*
sent_type	O	*exclaim, question, assert*	*exclaim* for clauses that are exclamations, *question* for those that are questions, and *assert* for all others	*assert*

Table 13
Features representing a clause in the coarse-grained data set.

Name	Category	Possible values	Description	Default value
char_in_clause	C	*yes, no*	*yes* if the clause contains a mention of a character	*no*
is_subj_obj	C	*yes, no*	*yes* if the clause contains a mention of a character and its grammatical function is subject or direct object	*no*
modified_by_np	C	*yes, no*	*yes* if the mention of a character is present in the clause and it is modified by a noun phrase	*n/a*
nbr_after_ first_mention	C	continuous	an integer that reflects the difference between the index of the current sentence and the sentence where the character is first mentioned (only defined for clauses containing mentions of characters)	*-1*

Table 13
(*continued*)

Name	Category	Possible values	Description	Default value
loc _in_prep	L	*yes, no*	*yes* if the clause contains a mention of a location embedded in a prepositional clause	*no*
loc_present	L	*yes, no*	*yes* if the clause contains a mention of a location	*no*
default_aspect	A	*state, activity, accomp, achieve*	default lexical aspect of the main verb in the clause; computed according to the privative model defined in (Dorr and Olsen 1997)	*n/a*
has_modal	A	*yes, no*	*yes* if the clause contains a modal verb	*no*
past_perfect	A	*yes, no*	*yes* if the clause is realized in past perfect tense	*no*
politeness _with_be	A	*yes, no*	*yes* if the clause contains one of the following expressions: *to be sorry, to be delighted, to be glad, to be sad*; the feature is designed to help capture politeness expressions (e.g., *I am glad to see you*)	*no*
simple_past _present	A	*yes, no*	*yes* if the clause is realized in simple present or past tense	*no*
tmp_exp _long_duration	A	*no, long, short*	*long* if the clause contains a temporal expression denoting a long period of time, *short* if it contains an expression denoting a short period of time and *no* otherwise	no
is_assertive _clause	O	*yes, no*	*no* if the clause is not an assertion	*yes*
is_assertive _sent	O	*yes, no*	*no* if the parent sentence is not an assertion	*yes*
nbr_of_sent	O	continuous	the index of the parent sentence in text	*-1*

Acknowledgments

We are grateful to Connexor Oy and especially to Atro Voutilainen for permission to use the Connexor Machinese Syntax parser free of charge for research purposes. We thank John Conroy and Judith Schlesinger for running CLASSY on our test set, and Andrew Hickl for doing it with GISTexter. Ana Arregui helped us recruit students for the evaluation. Many thanks to the annotators, summary writers, and raters, who helped evaluate our summarizer. A special thank-you goes to the anonymous reviewers for *Computational Linguistics* for all their incisive, insightful, and immensely helpful comments. Support for this work comes from the Natural Sciences and Engineering Research Council of Canada.

References

Artstein, Ron and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics (survey article). *Computational Linguistics*, 34(4):555–596.

Bartlett, Frederic C. 1932. *Remembering: A Study in Experimental and Social Psychology*. Cambridge University Press, London.

Barzilay, Regina and Kathleen R. McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–239.

Branavan, S. R. K., Pawan Deshpande, and Regina Barzilay. 2007. Generating a table-of-contents. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 544–551, Prague.

By, Thomas. 2002. *Tears in the Rain*. Ph.D. thesis, University of Sheffield.

Carenini, Giuseppe, Raymond Ng, and Adam Pauls. 2006. Multi-document summarization of evaluative text. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 305–313, Trento.

Charniak, Eugene and Robert Goldman. 1988. A logic for semantic interpretation. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, pages 87–94, State University of New York at Buffalo, Buffalo, NY.

Chawlar, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling techniques. *Journal of Artificial Intelligence Research*, 16:321–357.

Cohen, Jacob. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.

Comrie, Bernard. 1976. *Aspect*. Cambridge University Press, London.

Conroy, John M., Judith D. Schlesinger, and Diane O. O'Leary. 2007. CLASSY 2007 at DUC 2007. In *Proceedings of the Document Understanding Conference 2007*, New York. Available at `http://duc.nist.gov/pubs/2007papers/ida-umd.final.pdf`.

Cullingford, R. E. 1978. *Script Application: Computer Understanding of Newspaper Stories*. Ph.D. thesis, Department of Computer Science, Yale University.

Cunningham, Hamish, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. GATE: an Architecture for Development of Robust HLT applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, pages 168–175, Philadelphia, PA.

Díaz, Alberto and Pablo Gervás. 2007. User-model based personalized summarization. *Information Processing and Management*, 43(6):1715–1734.

Dorr, Bonnie J. and Mari Broman Olsen. 1997. Deriving verbal and compositonal lexical aspect for NLP applications. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 151–158, Madrid.

Dowty, David. 1979. *Word Meaning and Montague Grammar*. D. Reidel Publishing Company, Dordrecht.

Dyer, Michael G. 1983. *In-Depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension*. MIT Press, Cambridge, MA.

Elhadad, Noemie, Min-Yen Kan, Judith Klavans, and Kathleen McKeown. 2005. Customization in a unified framework for summarizing medical literature. *Journal of Artificial Intelligence in Medicine*, 33(2):179–198.

Fellbaum, Christiane. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Fuentes, Maria, Edgar Gonzàlez, Horacio Rodrígue, Jordi Turmo, and Laura Alonso i Alemany. 2005. Summarizing spontaneous speech using general text properties. In *Proceedings of International Workshop Crossing Barriers in Text Summarization Research, at Recent Advances in Natural Language Processing 2005*, pages 10–18, Borovetz.

Harabagiu, Sandra, Andrew Hickl, and Finley Lacatusu. 2007. Satisfying information needs with multi-document summaries. *Information Processing and Management*, 43(6):1619–1642.

Harkness, Janet. 1987. Time adverbials in English and reference time. In Alfred Schopf, editor, *Essays on Tensing in English, Vol. I: Reference Time, Tense and Adverbs*. Max Niemeyer, Tübingen, pages 71–110.

Huddleston, Rodney D. and Geoffrey K. Pullum. 2002. *The Cambridge Grammar of the English Language*. Cambridge University Press, New York.

Kazantseva, Anna. 2006. Automatic summarization of short stories. Master's thesis, University of Ottawa. Available at `www.site.uottawa.ca/~ankazant/pubs/thesis.tar.gz`.

Krahmer, Emiel, Erwin Marsi, and Paul van Pelt. 2008. Query-based sentence fusion is better defined and leads to more preferred results than generic sentence fusion. In *Proceedings of ACL-08: HLT, Short Papers*, pages 193–196, Columbus, OH.

Krippendorff, Klaus. 2004. *Content Analysis. An Introduction to Its Methodology.* Sage Publications, Thousand Oaks, CA.

Landis, J. Richards and Garry G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Lappin, Shalom and Herbert Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.

Leach, Chris. 1979. *Introduction to Statistics. A Nonparametric Approach for the Social Sciences*. John Wiley and Sons, New York.
Leake, David. 1989. Anomaly detection strategies for schema-based story understanding. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 490–497, Ann Arbor, MI.
Lehnert, Wendy G. 1982. Plot units: A narrative summarization strategy. In Wendy G. Lehnert and Martin H. Ringle, editors, *Strategies for Natural Language Processing*. Erlbaum, Hillsdale, NJ, pages 375–414.
Lin, Chin-Yew. 2004. ROUGE: A package for automatic evaluation of summaries. In Marie-Francine Moens and Stan Szpakowicz, editors, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona.
Lin, Chin-Yew and Eduard Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th Conference on Computational Linguistics*, pages 495–501, Morristown, NJ.
Lin, Chin-Yew and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In Marti Hearst and Mari Ostendorf, editors, *HLT-NAACL 2003: Main Proceedings*, pages 150–157, Edmonton.
Mandler, George. 1987. Determinants of recognition. In E. van Der Meer and J. Hoffman, editors, *Knowledge-Aided Information Processing*. North Holland, Amsterdam.
Mani, Indejeet. 2001. *Automatic Summarization*. John Benjamins B.V., Amsterdam.
Marcu, Daniel. 2000. *The Theory and Practice of Discourse Parsing and Summarization*. The MIT Press, Cambridge, MA.
McDonald, Ryan. 2006. Discriminative sentence compression with soft syntactic evidence. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 297–305, Trento.
Mei, Qiaozhu and ChengXiang Zhai. 2008. Generating impact-based summaries for scientific literature. In *Proceedings of ACL-08: HLT*, pages 816–824, Columbus, OH.
Merlo, Paola, Suzanne Stevenson, Vivian Tsang, and Gianluca Allaria. 2002. A multilingual paradigm for automatic verb classification. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 207–214, University of Pennsylvania, Philadelphia, PA.
Mihalcea, Rada and Hakan Ceylan. 2007. Explorations in automatic book summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 380–389, Prague.
Moens, Marie-Francine. 2007. Summarizing court decisions. *Information Processing and Management*, 43(6):1748–1764.
Nenkova, Ani and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The Pyramid Method. In *Proceedings of the Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics Annual Meeting*, pages 145–152, Boston, MA.
Nomoto, Tadashi. 2007. Discriminative sentence compression with random conditional fields. *Information Processing and Management*, 43(6):1571–1587.
Norvig, Peter. 1989. Marker passing as a weak method for text inferencing. *Cognitive Science*, 13(4):569–620.
Propp, Vladimir. 1968. *Morphology of the Folk Tale.* Indiana University Press, Bloomington, IN, 2nd edition.
Quinlan, J. Ross. 1992. *C4. 5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.
Radev, Dragomir and Daniel Tam. 2003. Summarization evaluation using relative utility. In *Proceedings of the 12th International Conference on Information and Knowledge Management*, pages 508–511, New York, NY.
Radev, Dragomir R., Simone Teufel, Horacio Saggion, Wai Lam, John Blitzer, Hong Qi, Arda Çelebi, Danyu Liu, and Elliott Drabek. 2003. Evaluation challenges in large-scale document summarization. In Erhard Hinrichs and Dan Roth, editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 375–382, Sapporo.
Rath, G. J., A. Resnick, and T. R. Savage. 1961. The formation of abstracts by the selection of sentences. *American Documentation*, 2(12):139–143.
Reeve, Lawrence H., Hyoil Han, and Ari D. Brooks. 2007. The use of domain-specific concepts in biomedical text summarization. *Information Processing and Management*, 43(6):1765–1776.

Rumelhart, David E. 1975. Notes on a schema for stories. In Daniel G. Bobrow and Allan Collins, editors, *Representation and Understanding. Studies in Cognitive Science*, pages 221–237. Academic Press, New York.

Salton, Gerard, Amit Singhal, Mandar Mitra, and Chris Buckley. 1997. Automatic text structuring and summarization. *Information Processing and Management*, 33(2):193–207.

Schlesinger, Judith D., Dianne P. O'Leary, and John M. Conroy. 2008. Arabic/English multi-document summarization with CLASSY - The past and the future. In *Computational Linguistics and Intelligent Text Processing, 9th International Conference, CICLing 2008*, pages 568–581, Haifa.

Scott, William. 1955. Reliability of content analysis: The case of nominal scale coding. *Public Opinion Quarterly*, 19(3):321–325.

Siegel, Eric V. 1998a. Disambiguating verbs with the WordNet category of the direct object. In *Usage of WordNet in Natural Language Processing Systems Workshop*, pages 9–15, Université de Montréal.

Siegel, Eric V. 1998b. *Linguistic Indicators for Language Understanding: Using Machine Learning Methods to Combine Corpus-Based Indicators for Aspectual Classification of Clauses*. Ph.D. thesis, Columbia University, New York.

Siegel, Sidney and John. N. Castellan, Jr. 1988. *Nonparametric Statistics for the Behavioral Sciences*. McGraw Hill, Boston, MA.

Soricut, Radu and Daniel Marcu. 2007. Abstractive headline generation using wild-expressions. *Information Processing and Management*, 43(6):1536–1548.

Tapanainen, Pasi and Timo Järvinen. 1997. A non-projective dependency parser. *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 64–71, Washington, DC.

Teufel, Simone and Marc Moens. 2002. Summarizing scientific articles: Experiments with relevance and rhetorical status. *Computational Linguistics*, 28(4):409–445.

Thorndyke, Perry W. 1975. *Cognitive Structures in Human Story Comprehension and Memory*. Ph.D. thesis, Stanford University.

van Dijk, Teun A. 1980. *Macrostructures. An Interdisciplinary Study of Global Structures in Discourse, Interaction, and Cognition*. Laurence Erlbaum Associates, Hillsdale, NJ.

van Dijk, Teun A. and Walter Kintsch. 1978. Cognitive psychology and discourse: Recalling and summarizing stories. In Wolfgang U. Dressler, editor, *Current Trends in Textlinguistics*. Walter de Gruyter, New York, pages 61–79.

van Halteren, Hans and Simone Teufel. 2003. Examining the consensus between human summaries: initial experiments with factoid analysis. In Dragomir Radev and Simone Teufel, editors, *HLT–NAACL 2003 Workshop: Text Summarization (DUC03)*, pages 57–64, Edmonton.

Vendler, Zeno. 1967. *Linguistics in Philosophy*. Cornell University Press, Ithaca, NY.

Vieira, Renata and Massimo Poesio. 2000. An empirically based system for processing definite descriptions. *Computational Linguistics*, 26(4):539–593.

Zechner, Klaus. 2002. Automatic summarization of open-domain multiparty dialogues in diverse genres. *Computational Linguistics*, 28(4):447–485.