Time Frames: Temporal augmentation of the news

by D. B. Koen
W. Bender

Great strides have been made in the use of computer tools to create, edit, filter, and present information, particularly since the tremendous growth in the mainstream popularity of the World Wide Web. The presence of a computationally rich environment at all stages of news distribution provides a unique opportunity to use these tools to improve the reader experience. Information provided for a general audience from a general source can be combined with small amounts of information specific to a reader to improve the reader's understanding of, connection to, and engagement with the news. This paper discusses Time Frames for extracting time information from news articles. Combining this time information with limited information about the reader, we explore the possibilities for improving the reader experience by augmenting news articles.

Most of us have a good ability to grasp in our heads the units and quantities of time that are used frequently in everyday life. If someone tells you that he or she will do something for you in an hour, you understand clearly the distinction between this and being told that the task will be completed in a week. Usually a person's conception of what constitutes a reasonable amount of time for completing a task coincides with that of others. In the event of a discrepancy, it is easy to work out the reason for the difference of opinion, which is probably based not on confusion regarding the units of time, but on the perceived difficulty of the task.

The simplicity of this example belies a hidden complexity in understanding temporal relationships. We have a good sense of how long a task that we perform frequently should take, but how well do we understand time frames that are outside of our daily personal experience? If we read in the newspaper that the House of Representatives of the United States spent a week deliberating on the country's budget for the coming fiscal year, should we be pleased, outraged, or should we nod and understand that this time is about right? The problem is that we do not know if this is a long time or a short time in that particular context.

What seems like a long time for some tasks seems like a short time for others. For events common to our experience we automatically understand where something falls in this range, but for events outside of our experience we do not. When individuals complain that Boston's "Big Dig"—a massive project to expand and move underground the central expressway through the city—is an inconvenience that seems to be taking longer than expected, do they understand how the project's 20 years of planning and construction (to be completed in 2004) compare to other similar projects? Does 20 years seem like a long time for a construction project in the downtown part of a city? Twenty years is about a quarter of the life of the average American. Who will the Big Dig affect more—a child born on the day the project was begun or that child's parents?

In an often-quoted section of *Being Digital*,[1] Negroponte compares his ideal computer-human interface to a "well-trained English butler." An extension of this idea would cast the machine in the role of a teacher, or possibly a collaborator, who understands the extent of the reader's knowledge of a subject and knows which concepts he or she may find difficult. This teacher would know how best to present new information to maximize this particular reader's understanding of difficult concepts. Could such a system be built to assist a reader in understanding the examples given above?

It is not the objective of the Time Frames research to implement such a system in its entirety, rather it is to provide a road map for additional work by providing a limited-domain example of augmentation.

Background

Time was chosen as the domain for this research because the almost universal appearance of time references in news articles makes the work broadly applicable across articles covering a wide range of subjects. Time and time measurements are of particular interest for another reason as well: the concept of time is simultaneously common and taken for granted and rife with hidden subtlety. Time can invoke history, as far back as the origins of the universe, or, as in the phrase "our times," refer to a particular historical period. Time can refer to a measurement of an interval or indicate a rate or tempo. It can refer to a continuum or it can specify a unique point in that continuum indicated by the combination of a calendar and a clock. And these are by no means the only ways that the concept of "time" can be used, merely those most relevant to this work. Suffice it to say that time is a "loaded" domain.

The modern layperson's vocabulary has come to accept a range of time measurements from on the order of 65 million years (thanks to filmed entertainment) down to the nanosecond level (thanks to the mainstream commodification of computer memory modules).[2] But perceptions of time and timeliness worldwide, and even among different ethnic and socioeconomic groups within the United States, are varied.

Differences in the pace of life and the importance of scheduling can cause considerable culture shock, for example, to the North American visitor to Brazil.[3] Indeed, people were not able to make even reasonably precise subhour measurements of time until around 1700, and nothing resembling the modern wristwatch appeared until around 1850.[4–7] Here in the United States at the end of the 20th century, however, we are angry if others are late; we synchronize our watches for detailed tasks or to make sure we catch a train on time; and, most of all, we insist on knowing "when." Would anyone read an article in a newspaper if he or she was not convinced it was current? What good is the "news" if it is not the latest version of the story with the most up-to-date details? A tropical storm hit Florida—when? The president has announced that U.S. troops are being deployed for a new "police action"—when? Why does it matter when? It matters because it allows us to put these events in a context that is familiar to our own lives. It allows us to say where we were when President Kennedy was shot (if we were alive at the time). Because of its perceived universality, time provides us with a shared language, signposts, and markers, forming a bridge to our own history and between our own history and other events that shaped and are shaping the world around us.

Article augmentation. The application of computation and networks to the publishing process has changed the way that news is created, distributed, and consumed. New news technologies are improving the speed and accuracy of data-gathering in the field. The processes for writing and editing news articles have also been subject to change. Finally, news selection has become a consumer activity. Once a general news article is written and edited, published, and selected and displayed for the reader, the next step is to use computation to combine this content with personal user data, altering and extending the individual article with the intent to improve the reader's understanding of the material presented.[8] (Augmentation as it applies to searching is not discussed in this paper.[9]) Web sites such as those of the Cable News Network (www.cnn.com) and MSNBC (www.msnbc.com) have experimented with this notion of article augmentation, often including links to source research material, images, multimedia clips, discussion groups, old or current news articles on the same or similar topics, and related Web sites. For the most part, these augmentations are performed by a group of human experts.

Using computational tools to analyze the news is not a new idea.[10] An early example of machine-generated augmentation of the news is *NewsPeek*,[11] a personalized electronic newspaper developed in the early 1980s. Articles were automatically augmented with maps and images based on matching keywords

in a database of 20 000 news photographs and graphics provided by the Associated Press.

Elo's *Peace, Love, and Understanding Machine* (PLUM)[12,13] is a more recent example of computationally generated augmentation of the news. PLUM uses augmentation to localize the news by associating with the reader only one piece of information—the location of his or her hometown. Working with a corpus of articles about natural disaster, PLUM provides analogies to the reader's local community or country to help him or her understand the significance of things like death tolls, financial losses, and property damage. It also helps put the disaster into historical context for both the affected region and the reader's region.

Warren Sack's *Questioning News*[14] provides another example of the modification of existing news articles in order to engage the reader. The Questioning News system "automatically annotates news articles with open-ended leading questions" in an attempt to encourage the reader to ponder these questions and, it is hoped, to be more likely to understand and remember the article.

These last two projects demonstrate the potential for providing augmentation that is not just somehow related to the core article, but actually enhances its accessibility and utility.

Time Frames

Time Frames is similar to the handful of previous systems that automatically provide supplementary information for an individual article to help the reader connect to and understand the material. Whereas some of these systems, such as PLUM, operate on a *narrow domain* of the news, Time Frames attempts to be applicable to news in general. Time Frames builds augmentations by using time information, which is found in some form or another in almost *all* of the news. Other systems, such as Questioning News, although of use for examining individual articles, do not create a structure that can be used to put the article in an external context. Time Frames, by extracting temporal features, is able to provide an external reference point: time.

With Time Frames, the means of augmenting news articles to improve the reader's experience with an article are explored, using references to time extracted from the article. These augmentations fall into five categories: (1) providing personal context, (2) providing local context, (3) providing historical context, (4) providing alternative visualizations, and (5) providing tools to question the content of news articles.

Personal context. When hearing or reading about events from the past we often find that we can connect better to the story if we can place it in the context of what we were experiencing at the time of the story. For example, when recently told a story about the original introduction of the NeXT** computer in early 1989, a student remarked:

> I paused for a moment to put this in the context of my own experience—I remembered that I was beginning the second term of my freshman year at MIT when our first NeXT computer arrived at the Media Lab. Not only did this personal relationship to the story connect me to my experiences with NeXT computers, but it also allowed me to easily recall the computing environment that was available, the software and systems that were in use at the time, and the projects that I and others were working on at the Lab during that period.

This anecdote is only intended to be an example. It is beyond both the scope and the spirit of this research to attempt to model the entire life of a reader in order to provide a personal anecdote about any time reference in a news article.

The goal of Time Frames is to do as much as possible while minimizing the requirement to collect new information from the reader—to keep in check the desire to build large knowledge bases to provide sophisticated augmentations. Such systems may well prove valuable, but the spirit of this research is to suggest how easy it is to leverage the information that is easy to obtain, or already accessible to us, in order to provide interesting and meaningful augmentations.

Local context. Providing local context is similar to providing personal context. The distinction is in the addition of a new piece of knowledge to the system—a location with which the reader identifies. The connection to a place and the events that occurred in that place can form a powerful link to the past. For example, in the anecdote given earlier, an integral part of the memory was remembering where the student was at the time—MIT. Reminding a reader of events in his or her hometown can, just as with personal context, bring back a feeling of connection to the events described. Instead of "What

were you doing," the context is "What were the people around you doing? Do you remember when that big tornado touched down in a residential district?

Providing historical context for news is considered from three different perspectives in Time Frames.

That is the time frame to which this news article refers. Let me go on to explain what Saddam Hussein was doing at that time in Iraq."

Historical context. Providing historical context for news is considered from three different perspectives in Time Frames. The first is to globally contextualize a point in time. The second is to show a point in time from the historical perspective of different locations. Finally, the third is to take a single theme and analyze that theme through time.[15] By globally contextualizing a point in time, Time Frames is able to provide the reader with more information about other events that occurred at the time referred to in a news article. This does not assume that Time Frames is able to provide a specific piece of information for which a reader might be looking. It is intended to facilitate serendipitous encounters with news articles a reader might not otherwise have noticed, were it not for the temporal connection. Time Frames attempts to tailor the historical perspective for the reader.

Many of the time references contained in news articles are references to prior events to establish the historical basis for the events in the article. Ideally this history should be at the correct level of detail for the reader, but because news articles are written for a generalized audience, the journalist has to make a best guess as to the level on which to provide the context. If some or all of this information were provided at the point where the news is packaged for the individual, it could be more relevant and topical for the reader. The third type of historical context augmentation explores the changes in thinking about a particular topic over time.

An example of providing a historical perspective on something current is shown in a research project done by the MIT Media Laboratory's Explanation Architecture Group. The idea of the work-in-progress Image Maps project[16] is to allow a user to take a photograph with a specially modified camera and then show the user historical photographs taken from the same location and perspective. For the purposes of Image Maps it is sufficient to find any photograph from the same location. Because of the abundance of news and information about history available, it is not sufficient to pick just any article from another time period, of course. It must not only be vaguely related to the current article, but must, in fact, be demonstrably on the same topic. Otherwise readers will most likely consider it irrelevant and will probably begin to question why their time is being wasted with extraneous information.

Alternative visualizations. Another type of augmentation explored in this research is the idea of giving the reader a new perspective on information that is already available in the news article. Different people process information in different ways, and sometimes a new perspective on the same data can help a reader to better grasp relationships, find new connections, and comprehend the information presented. Three different ideas are described, based on the same basic concept, to give the reader a more global look at time information in the news.

Questions. Sack's Questioning News system demonstrated the value of using machine-generated questions to engage the reader in a news article while at the same helping the reader to learn and retain more of the information. In Time Frames the intent is to use machine-generated questions to provide the reader with opportunities to explore beyond the information presented in the article.

Infrastructure

The first objective is to extract time information from news articles. Figure 1 shows an example of the types of information needed to be reliably extracted from the body of an article. This section describes a robust system for extracting time information from news articles, then parsing and massaging this raw data into a usable form. To solve the time extraction problem, a reusable Perl module was implemented and tested.

Several natural language parsing systems were considered for this vital portion of the research.[17–21] However, because the Time Frames system is intended for a specific purpose (i.e., time extraction),

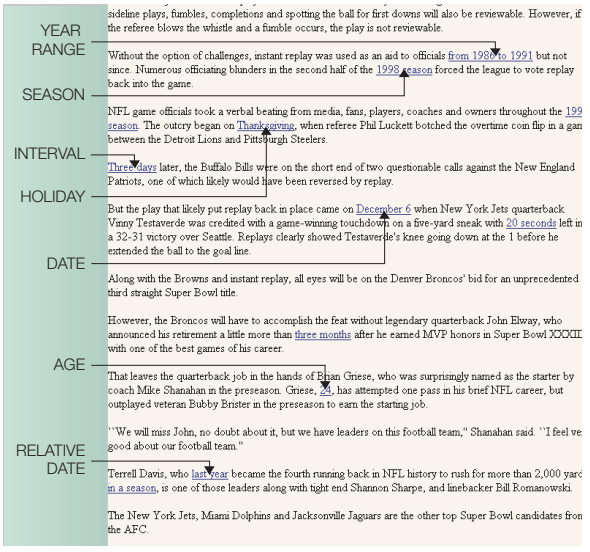
the general-purpose parsers tended to provide unneeded sophistication in many areas irrelevant to the problem and to provide either insufficient consideration or incomplete understanding of time extraction for the task at hand.

Because the needs for identifying and extracting information are domain-specific, it was possible to achieve excellent results without a full grammatical parse of the news. It should be acknowledged, however, that once the desired information is located, the additional information afforded by a full text parse could be used to improve understanding of its context.

Extracting. News articles are searched for possible date and time matches. These components are extracted as text strings. The broad categories that the system searches for are: intervals, ages, and datetimes. Some additional processing is needed to properly match and categorize these strings. For example, a complete UNIX** time reference (Thursday August 12, 1999 at 10:32 AM EDT [Eastern Daylight Time]) contains a valid day-of-the-week word, date, and time stamp. Each of these fragments could constitute a valid match by itself. If each was to appear in isolation, the system would attempt to generate a more complete reference for them. However, the system favors matches that are more specific over matches that are less specific in order to make sure that all relevant components of a temporal reference are incorporated. In most cases, this simply means aggregating the longest possible sequence of subelements. In some cases, it requires recognizing when two contiguous substrings are not part of a single reference, as in the following example: "It was already Tuesday—January 3, Wednesday, was when the applications were due. I would never make it on time." Thus, multiple passes are needed to disambiguate the wide variety of both ordering and delimiting of the subelements of temporal references.

Intervals and velocity. The system can extract intervals in units of centuries, decades, seasons, years, months, weeks, weekends, days, hours, minutes, and seconds in various long and short forms. The value of the interval can be expressed as a number or written out as words and can be delimited using any nonalphanumeric characters. This form also supports using the word "a" instead of the number "1." Intervals can also be expressed as a range. This form allows the insertion of a connector, either "to" or a hyphen ("-"), and a second number after the connector and before the units, for example "twenty to twenty-five

Figure 1 A sample news article



minutes." This same matching pass extracts velocity in miles per hour, kilometers per hour, meters per second, and knots. Velocity, being a measurement of distance per time unit, is included for completeness. Some examples of this pattern are: "2 weeks," "1 weekend," "a year," "twenty-thirty years," "twenty to thirty years," "13 yrs," "2 wks," "250 kph," and "two-hundred fifty miles per hour."

Age. Ages are extracted in three different ways. First, the system searches for strings such as "29-year-old," "15 years old," and "three year old." In order to avoid a second search for a largely similar expression, this type of age is actually extracted with the same basic pattern as the intervals described above. The interval pattern is extended to include an optional "old" at the end. If this optional portion matches, the string is considered an age rather than an interval. This allows the age matcher to have the same features as the interval matcher—the pieces can be delimited using any nonalphanumeric character, and both numbers and words representing numbers can be used. Age is not restricted to years, but can use any of the units used for interval matching.

The second age extraction pattern was added after it was determined in testing that, while it does occur, the first pattern is not the most common in news articles. It is more common to indicate the age of a person in a news article after the person's name,

delimited by commas, such as "John Smith, 30, ..." To avoid collision with two-digit years, ages are not matched unless they are preceded by a comma and white space and followed by a comma. Also, ages detected in this way must be one or two digits. A third, simpler age pattern was included for occasions in which an age is specified explicitly in the form "age 22." Ages of this type were also required to be one or two digits. Some examples of the age patterns are: "29-year-old," "15 years old," "three year old," ", 22," and "age 22."

Dates and times. "Datetimes" are the largest and most extensive category extracted by the time extractor module. A datetime is text that is resolvable by a human to a specific date, time, or range of dates or times. The remainder of this section describes the different subtypes of datetime matches and how they are detected.

Centuries and decades are extracted first. A century match is detected on any four-digit number ending with the string "00s." Some examples of the century pattern are: "1800s" and "1900s." A decade is any four-digit number in which the first digit is 1–9, the second 0–9, the third 1–9, and the fourth 0, followed by a trailing "s." Some examples of the decade pattern are: "1980s" and "1890s."

A "plain date" is a time-free date reference with an optional leading day of the week and one of the following: (1) a month (word form only), a day followed by an optional "st," "nd," "rd," or "th," and a year; (2) a month (word form only) followed by a year (in either two- or four-digit form; (3) a month (again, word form only) and a day; or (4) an inverted date—that is, the day of the month first (word form or a number with an "st," "nd," or "th" appended), followed by the word "of," followed by the month (word form only), and an optional year. Some examples of the plain date pattern are: "December 1992," "December 31," "Thursday December 31, 1992," "Thursday December 31st, 1992," "23rd of July," and "Eighth of December."[22]

Because the domain of Time Frames is the news, a simple representation of a time stamp was not sufficient. In addition to the normal, common representations of time, we had to support various ways of including time in prose. To that end a "valid time" is defined. Any time may be preceded by an optional leading "at." The "at" must be followed by a time expressed in Greenwich Mean Time (a four-digit number followed by "GMT") or any of the following: (1) a time expressed in either 12- or 24-hour "colon" form, as in 9:15 or 13:01, (2) a number between 1 and 12, and (3) a time expressed as words (ten-thirty, six-fifteen, ten twenty-one, etc.). In the third case, the time may be preceded by "half," "quarter," "past," "of," "til," or "to" if the time is a whole hour. In all three cases one or more of the following optional additions may follow the time: (1) the word "o'clock" with or without an apostrophe and in upper- or lowercase, (2) an "AM" or "PM" designation, with or without a leading space, with or without periods, and in upper- or lowercase, (3) any of the following: "in the morning," "in the afternoon," "in the evening," "at night," or (4) a time zone specified in any of these forms "ET," "EDT," "EST," "eastern." Time zone specifications are case-insensitive.

If the time is specified as a number between 1 and 12 or in words representing these numbers, then it must be followed by at least one of the five options just described. If the time is expressed in "colon" form, then zero or more of the options may be used. Military style times (0800 hours) are not currently supported. Some examples of the time pattern are: "9:15," "9:15 PM," "9:15 p.m.," "9:15 p.m. EST," "13:01," "13:01 EST," "0530 GMT," "9:15 in the morning," "six thirty-two p.m.," and "half past eight in the evening."

"Date and time" is the main representation of a fully expressed date and time. It consists of a plain date followed by a valid time. Some examples of the date and time pattern are: "Saturday January 4, 1992 12:15 pm," "January 4, 1992 at 6 o'clock in the morning."

A day of the week (e.g., "Monday," "Tue," etc.) may be preceded by a time of day in valid time format, optionally separated by the word "on." Some examples of this are: "10:15 pm Monday," "six thirty-five on Tue," and "midnight Friday." A day of the week may be followed by time of day. Some examples of this are: "Friday midnight" and "Tue at midnight."

A season is a year or a year range followed by the word "season." This is to capture references in sports articles, such as the "1998 season" or the "1997–1998 season." Seasons are also captured as interval units—such as "two seasons ago," as described previously.

A number followed by "th" and the word "anniversary" is considered a datetime because it is resolvable to a specific date for the original event. An example of this pattern is: "25th anniversary."

"Bare" years occurring in news articles are vital information, yet present some interesting problems. A year is, of course, simply a two- or four-digit number. Two-digit numbers occur very frequently in news articles, so they are not flagged as dates unless they are preceded by an apostrophe. Four-digit numbers can occur in news articles for a variety of reasons. But four-digit years are important enough that they cannot be eliminated, so an effort is made to reduce "false positives." For example, a four-digit number is eliminated as a year if it is followed by any amount of white space and then another digit. This eliminates such things as stock and index quotes. Other false hits had to be explicitly identified in order to be removed—such as telephone numbers and Social Security numbers.

Year ranges are a completely different pattern from bare years because the requirement for two different years (the range boundaries) and an explicit specification of range eliminates ambiguity. A year range is identified as two four-digit numbers separated by a dash ("–"), with the pattern "between [four-digit number] and [four-digit number]," or with the pattern "from [four-digit number] to [four-digit number]." Some examples of these patterns are: "1992–1994," "between 1992 and 1994," and "from 1992 to 1994."

The *n*th day of the week in month form matches an ordinal identifier expressed either as words or as a number followed by "st," "nd," or "rd"; then a day of the week; then the word "in" followed by a month. An optional year may follow the form; then an optional "at" followed by a time. Some examples of this pattern are: "third Friday in June," "2nd Saturday in July," and "fourth Friday in June 1992 at 4:15 PM EST."

The last/next day of the week form is the word "this," "last," or "next" followed by a day of the week ("Monday," "Tue") and an optional time. Some examples of this pattern are: "This Saturday at 12:30," "Last Tue," "Last Tuesday at 4:30 PM EST," and "Next Friday at 4:30 PM Eastern."

The main pattern for units into the future is the word "in" followed by a number specified either in digits or in words such as "in two weeks." Ranges are also supported here, and can be expressed using digits or words, and separated by a dash ("–"), the word "or," or the word "to." Various options can be added to this primary form to make it more specific. For example, in one form the "in" may be preceded by an optional day of the week and an optional time of day, for example "Friday in two weeks at 10:20 pm." Another allowed form would add an optional day of the week and optional time after the main pattern, such as "in two weeks on Friday" or "in a week on Friday at 8:00 am." The last example also demonstrates that all of the patterns that express numbers as words allow the use of the word "a" in place of "one." Some examples of the future pattern are: "in two weeks," "in two or three weeks," "in two to three weeks," "in 2–3 weeks," "in 4 days," "Friday in two weeks at 10:20pm," "in two weeks on Friday," and "in a week on Friday at 8:00 PM EST."

Units into the past are analogous to the pattern for units into the future, but instead of the primary pattern being exemplified by "in two weeks," it is exemplified by "two weeks ago." The modifications to the basic pattern remain the same as in the future references. Some examples of this pattern are: "two weeks ago," "4 days ago," "Friday two weeks ago at 10:20pm," "two weeks ago on Friday," and "a week ago on Friday at 8:00 PM EST."

Time of day can be expressed in words—supported references are "morning," "noon," "afternoon," "evening," "night," and "midnight." This basic pattern can be modified with an optional "mid," "late," or "early"; an optional "this," "last," or "next"; or an optional "today," "yesterday," or "tomorrow." Some examples of this pattern are: "morning," "mid-afternoon," "this evening," "yesterday afternoon," "early yesterday morning," and "late this morning."

The last/next form captures references to "this," "last," or "next" of any time unit using the same units as the interval patterns. Some examples of this pattern are: "this weekend," "next year," and "last decade."

The strings "today," "yesterday," and "tomorrow" are found alone or followed by the string "at" and a time of day in valid time format. Some examples of this pattern are: "today," "today at 12:30pm," "today at 13:01," and "today at noon."

As subcomponents of the date forms, months can be expressed as either words or numbers. For "bare" months only words are allowed, but months may be expressed in either a long or a short word form, such as "January" and "Jan." As with all other patterns in the time extractor, bare months are case-insensitive with a few exceptions. Capitalization of the months "May" and "March" (and short form "Mar") is enforced to avoid false hits with the words "may,"

"march," and "mar"; and the short form of April, "Apr," must not be in all caps, to avoid false hits on references to credit card annual percentage rates (APR).

The time extractor recognizes named holidays, including "Christmas," "Christmas Day," "Christmas Eve," "Easter," "New Year's," "New Year's Eve," "New Year's Day," "Thanksgiving," "Rosh Hashanah," "Yom Kippur," and "Halloween." An optional string "season" may follow each of the above. Although in some cases adding the word "season" does not make sense, the objective of the patterns at this stage is to locate possible matches; it is not the role of the time extractor to determine the rationality of the date and time references in the news articles.

Days of the week are recognized without additional specification of time or date.

Massaging and parsing. After extracting datetime references from the text, it is necessary to convert them to a normalized date form whenever possible, so that they can be compared to each other and sequenced. Some of the references need to be manipulated a bit before attempting to parse them. References to units into the past and future have to be converted into a consistent form. For example, "a week ago" is converted to "−1 week" and "in two weeks" is converted to "+2 week," then parsed relative to the current day. The years are extracted from year ranges and converted to a normalized form to be parsed separately. As already mentioned, most matches are case-insensitive, but a few exceptions must be made to avoid mismatches on month and day-of-the-week abbreviations. Telephone numbers, Social Security numbers, and any matches followed by a percent sign or the word "percent" are rejected here.

The most complex manipulation done at this stage is an attempt to determine if bare day-of-the-week and month references are referring to the past, present, or future. The system first looks forward and backward from the reference to find the extents of the sentence, using simple heuristics to avoid false sentence endings (such as "Mr." and "Gen."), and imposing constraints on how far to go from the reference to avoid excessive captures of undelimited text.

After the sentence containing the reference is located, it is searched for words that appear to be references to the past or the future. This is done by matching the regular form of past tense verbs (ending in "ed," but eliminating frequently occurring mismatches such as "red" and "bed") and commonly used irregular past or future tense verbs. Once this information is extracted from the sentence, it is not completely clear how to use it—many sentences in a news article contain references to both the past and the future.

We assume that if the sentence contains any references to the past then the system should place the reference in the past. This technique is not always effective, of course, and fails spectacularly when past tense verbs are used in a quote. Our objective was to make a best guess while keeping the system as efficient as possible and, to guarantee its extended usefulness, to avoid introducing dependencies on external systems. Future work on this module might include using a more sophisticated system for determining whether references such as bare days of the week are intended to be references to the past, present, or future. After the text references are converted to a form that the parser can understand, and are parsed to a normalized date form, the system attempts to prioritize overlapping references.

Most overlaps are eliminated in the pattern-matching phase by careful choice of the ordering of patterns in the datetime search pass. Some overlaps still occur with intervals and datetimes, however. For example "in two weeks" is a valid datetime reference, but contains within itself a completely valid interval ("two weeks"). Because intervals and datetimes are of different types and are matched separately, both will be included in the list of time terminology found in the article. Because the module is intended to be general-purpose, the elimination of overlaps occurs in a separate stage from the extraction and parsing. For purposes of this research, however, the overlap elimination strategy is straightforward—the longest pattern is kept. This will tend to favor datetimes over intervals. Because the "two weeks" in "in two weeks" is a valid interval, however, it is conceivable that for other purposes, an interval would take precedence over a datetime.

The time extractor normalizes extracted time references relative to "today." Many of the time references in news articles are expressed relative to an assumed date of publication. In order to support parsing databases of stored news, or any other block of text for which the concept of "today" does not refer to the current day, the time extractor supports

a mechanism for specifying "today" as an arbitrary date. In the case of news articles this will usually be the publication date or, if publication date is unavailable, the transmission date of the article.

Testing. The patterns for the time extractor were originally designed using carefully crafted sample blocks of text, demonstrating each of the forms to be extracted. Once the system was able to properly match these sanitized inputs, it was tested against wire-service news items from the Associated Press and Reuters. The tests included 520 articles from nine categories: top stories, business, technology, politics, world, entertainment, sports, science, and health. The time extractor found 4426 time references in the test set. Early testing was accomplished by quick visual inspection of the results of the time extractor's efforts. Later, a testing utility was built to speed up this process and to help generate a more complete picture of the time extractor's accuracy. The testing utility still required manual visual inspection by a human, but greatly simplified the process by the use of a World Wide Web CGI (common gateway interface). This interface kept track of which articles had already been scanned and always presented the tester with a new article, even across sessions. Results were automatically recorded and tallied.

For this research it was deemed more important that the matches be precise than that they be complete. To this end certain types of potentially valid values were eliminated because they were determined to frequently generate false matches in news articles. For example, unless embedded in a complete date string or preceded by an apostrophe, years with the century left implicit are rejected. For example, "89" is rejected, but "'92" is allowed. Yearless dates expressed with numeric months and month and year combinations expressed with numeric months (e.g., "12/31" for "December 31" and "01/01" meaning "January 01" or "January 2001") were also eliminated, both because of frequent mismatching and because of ambiguity as to which meaning was intended. While originally considered theoretically valid date forms by the pattern matcher, dates of this form were never seen in any of the test data, so their elimination was considered unlikely to have any effect as long as the input remains constrained to news articles. Table 1 shows the frequency in the test set of each type of time information extracted.

Evaluation. Because a large portion of this research was intended to determine the feasibility of extracting time information from news articles in a general way, each stage of the time extractor's work was implemented independently, without consideration for the feasibility of using the data collected in the next stage. This was done to ensure the maximum possible reusability of each component. It should also be noted that when implementing the time extractor, no consideration was given to whether or not an experiment had been designed that specifically required that information—all possible time information was identified and extracted. This effort will allow the module to be used in future experiments with minimal modifications.

Table 1 The frequency of each type of time information extracted from the test data set

	Number in Data Set	Percent of Total
Intervals and velocity	771	17.42
Ages	199	4.50
Dates and times	3456	78.08

Table 2 The recall and precision of the time extractor

	Recall (%)	Precision (%)
Intervals and velocity	89.36	91.27
Ages	59.70	95.00
Dates and times	93.85	87.50

Extraction. The success of the extraction layer for each type of time reference is shown in Table 2. *Recall* is the ratio of valid matches found by the extractor to the valid time references in the data set. *Precision* is the ratio of valid matches found by the extractor to total matches found by the extractor. These values were determined by manual inspection of 108 randomly selected articles from the entire test set. The overall recall of the evaluated set was 89.93 percent, while the overall precision was 88.64 percent.

Recall. Recall was reduced in many cases by a conscious decision not to extract certain types of time references because of their imprecision. In other cases recall was reduced by the decision to sacrifice recall for improved precision. In still other cases, the testing exposed the system's incomplete coverage of particular types of time references. Examples of imprecise time references that the system was never coded to detect, but that were included in the recall

analysis, are references such as "earlier," "later," "during the day," "a short while ago," "within hours or days," "in recent days," and "hours before." This kind of reference is not reducible to a specific point in time or even to a specific range, therefore these references were not included for extraction.

If a more sophisticated method of representing and working with time were put in place, these vague references might yield some useful information. In some cases recall of years was sacrificed in order to improve precision. For example, one reviewed article made reference to the years 100 and 200. The decision was made to improve precision by extracting year references of only four digits, or two digits with a leading apostrophe. Relaxing these constraints resulted in increased false hits. As described elsewhere in this paper, other methods were used to improve year precision. For example, four-digit numbers followed by any number of nonalphanumeric characters, in turn followed by a digit, were eliminated from consideration. Including these values resulted in dramatically reduced precision because of stock quotes, currency conversions, and other decimal or otherwise delimited numbers. As a result of always eliminating these patterns, year recall suffered, but precision was improved.

Testing also exposed the system's incomplete understanding of some types of data that show low recall. Ages in particular have a low recall. This was the result of two factors. The first is that ages appearing in a news article without clear identifying words surrounding them cannot easily be detected with high precision, because an age is just a number. Identifying two-digit numbers surrounded by commas yields high precision when identified as an age match, so this form was included in the age pattern. Other ages without identifying text, such as "He is 12," result in too much loss of precision. The second factor has to do with some forms of ages found during testing: "teens," "twenties," "under 25," and "aged 15–25." These forms were not included in the time extractor because they are not resolvable to a specific age, unlike the three forms that the time extractor does detect. Extending the system to detect the types of age references that reduced the recall would require a corresponding change to the way that we model ages—to allow for age references that are ranges instead of specific ages.

Season information ("summer," "winter," etc.) is not included. Very few holidays are currently supported. It should be noted, too, that during the tests the term *holiday* was expanded to include any "named day"—for example, "Election Day." There was no category at all for some types of data found in the tests. For example, the "fourth quarter," referenced in a financial article, could easily have been in a sports article as well, though this particular example did not occur in the tests. Recall for centuries was very low. The human recall analysis exposed a new pattern for centuries: "15th-century." This was a form that had not been seen in the training set, which included only the form "1500s." Having missed this form is somewhat excusable considering the low incidence of century data in both the training and test data sets (only two appear in the entire test set; one detected and one missed).

Precision. Several different types of errors resulting in reduced precision were detected during testing. Articles that reference stopwatch or game time are always assumed to refer to clock time. For example, in the sentence "Trailing 22–19 with 4:08 to play, the Wolverines staged a final drive . . . ," the segment "4:08" is assumed by the time extractor to refer to 4:08 AM today. Similarly, in the sentence "Ridden by Gary Stevens, Hollycombe covered one mile in 1:35 for his fourth win . . . ," the segment "1:35" is assumed by the time extractor to refer to 1:35 AM today. This type of mismatch occurred 27 times in the test set, though it should be noted that the mismatches were concentrated in only a few articles. Because not a single article in the entire test set contained a clock reference without any indication of the time of day, this mismatch could easily be fixed by modifying the pattern to require a time of day designator.

A single mismatch instance was found in the test set in the following sentence: " . . . opened membership talks with five eastern European hopefuls . . . ," in which the time extractor interpreted "five eastern" to refer to 5:00 AM eastern time. Nowhere in the tests was this time form ever used to refer to a time of day. Another mismatch occurs when a number directly precedes a day of the week: " . . . as the Texas Rangers beat the Chicago White Sox 8–6 Monday in the opener of . . . ." In this sentence the time extractor finds "6 Monday" and interprets this to mean 6 o'clock AM on Monday. This error is a side effect of all nonalphanumeric characters being treated equivalently by many of the patterns in the time extractor. This decision was made to allow for the variety of ways of delimiting times, dates, intervals, and ages. In this case, it prevents the time extractor from recognizing that the "6" is being pulled from a game score. This error is relatively uncommon, occurring

12 times in the sample set, and always in sports articles referring either to game scores or to player scoring records.

A variant of this precision error, with the number expressed in word form, was seen four times: once in an article about golf (" . . . fought through a sometimes strange and constantly competitive back nine Monday to hang on for a one-stroke victory at the Rail Classic"); once in an article about the college football team the Sun Devils: "The next two Sun Devils scores came after . . . ," which was mistakenly interpreted to refer to 2:00 AM on Sunday; and twice in a single sentence in an article about baseball: "Hammonds, who hit one Saturday and two Sunday." It should be noted that not once, in the entire test set, was either the number or the word form a correct match for a time of day followed by a day of the week. Based on the testing, this form should probably be eliminated in cases where the extractor is to be used only for news articles.

A mismatch also occurs in interval matches. Because an interval unit of one may be specified with the word "a," sometimes adjectives beginning with a unit of time measurement and preceded by the noun-identifier "a" will result in a false match, as in this sentence: " . . . said Jamie Wolcott, a year-round resident of . . . ." Also, unusual grammatical structures, usually found in quotations in news articles, can result in mismatches of this type: "I felt like today was as difficult a day as I've ever had . . . ." In this sentence "a day" is interpreted by the time extractor to refer to an interval of one day. All the interval mismatches resulted in clearly erroneous matches about 15 percent of the time. Finally, the use of "a" to mean "per" occurred relatively frequently in the test set: " . . . the company pays 30 million claims a year and only . . . ," but this usage is not completely inconsistent with the interpretation of it as a time interval. The form was used in the article with the same meaning expected by the time extractor 47 percent of the time, and the other 38 percent of the time the intended meaning was "per," which is arguably not an erroneous match.

By enforcing capitalization rules, an attempt is made to avoid mismatches with date and day-of-the-week words that have meanings in other contexts ("mar," "sun," "sat," "APR," "march"). Mismatches do occur when these words occur in article headlines and are therefore capitalized as are their time and date word counterparts. For purposes of generality, the time extractor makes no attempt to constrain four-digit dates to a range that "makes sense." As a result, the test set revealed four types of false matches on years. One occurred in a note outside the body of the article indicating the exchange rate of "($1–.6223 Pound)," a problem that occurred three times in the test set. The system currently checks to ensure that possible four-digit dates are not followed by a decimal point. An additional check to eliminate four-digit numbers preceded by a decimal point would fix this particular example. The second type of year-matching error would be remedied by the same fix as the first—fractional stock share prices with the portion after the decimal point displayed to precisely four digits. This second type of error was seen in two articles in the test set. The third incorrect match for a year occurs in addresses—in the test, the "1600" in "1600 Pennsylvania Avenue" was matched once. The final type of erroneous year match was, in this case, simply a four-digit number in an article referring to a possible problem of representing numbers in computers. Surprisingly, this error occurred only twice in the entire test set and, while difficult to fix, does not seem to represent a significant problem, because of its infrequency.

The name "Jan" is lexicographically indistinguishable from a reference to the month of January.

One year-range mismatch was found, in an article on the question of computers having difficulty with the year 2000: " . . . some might not be able to differentiate between 2000 and 1900." This error, in which "between 2000 and 1900" is interpreted to indicate a date interval, was seen only once. This particular example brings to mind a quick check that would solve at least some problems of this type—the candidate interval dates should be checked to ensure that they are in the proper order.

A few other precision errors were detected infrequently but exposed interesting problems with this type of extraction. For example, the term "night owl" was used in one article and the extractor was unable to determine that this usage of "night" was not actually intended to refer to the time of day in the context of the article. Another article referred to a

person named "Jan," which is lexicographically indistinguishable from a reference to the month of January. The system also matched on the word "Today" in the name of a television program *Early Today*.

Normalization. The only normalization that is done on intervals and ages is to separate the units and the values into distinct fields to be returned by the module. If the value is represented in words instead of digits, no conversion to digits is done. The stage that parses extracted datetimes into normalized times works on a subset of the data collected from the previous stage.

What the time extractor does not do. The time extractor does not guarantee that the dates it finds are rational. For example, if an article refers to "Thursday, September 19, 1997," the stage of the system that locates text strings will flag this as a date. Because September 19, 1997 was a Friday, however, the stage that attempts to convert the text strings into a normalized date form will fail. However, the reason for the failure is not reported back, because the date parser uses a series of patterns to find the pieces of the string that make up the date and the possible points of failure are many—it is difficult to distinguish a near match from a complete failure. Because the focus is on news articles, it was deemed unnecessary to modify the system to recognize the cause of these failures. News articles should rarely, if ever, contain dates that make no sense, and in the event that they do there is very little we can do to report this to the author, because our system is at the distribution and consumption point.

Some types, such as the year range, appear as an interval but can also be normalized to a specific date on each end of the range. For the purposes of this tool the latter interpretation was deemed more valuable and the interval match is not even detected. Future versions of the time extractor module could support multiple interpretations of a single match and thereby capture the interval nature of this match as well as the datetime interpretation. The time extractor does not maintain date and time context on a sentence, paragraph, or even story level. Each piece of information extracted from the body of text is treated independently. This means that when the time extractor converts certain extracted pieces of data into normalized date form it will give them the wrong value. For example, consider the following paragraph:

> On September 23, 1992 Mark woke up at his usual time of 6:30 AM. He ate breakfast, brushed his teeth, and then went for his morning jog. When he returned home at 7:30 his home had been broken into and his possessions stolen.

If this article's transmission date were any day other than September 23, 1992, the time extractor would have no way of knowing that the times, 6:30 AM and 7:30, were intended to refer to that day. Even properly setting the transmission date, as described earlier, would lead to errors if the article were referring to a historical event. In most cases it is safe to assume that bare time references are referring to the same day as the transmission date of the article. In this case, however, a date is clearly specified at the beginning of the paragraph and any human reader would clearly realize that the time references that follow are intended to be taken to occur on the date specified. Because the time extractor does not do a full parse of the article, but seeks to quickly extract small pieces of information based on patterns, it does not know to treat the times relative to the date specified. Future work on time extraction should probably take this possibility into account.

In addition to the date parser's inability to understand and properly handle certain types of time references that are extractable, some types of time references were not included in the time extractor at all. One notable example is that the extractor does not even extract information about seasons—as in "winter," "spring," "summer," and "fall." While an unfortunate omission, the research to date has not been hampered by the lack of this information. Its inclusion would, however, make an excellent addition to the module.

Experiments

Several augmentations are explained and proposed to meet the objectives of improving the reader's experience with an individual news article.

Personal context. To demonstrate personal context the time extractor was used to extract four-digit years from incoming news articles. The objective was to try to help remind the reader of the year referenced. The system uses three databases: the Internet Movie Database's database of Academy Award winners and nominees, the Internet Movie Database's database of world events, and a database of toy inventions.

Articles are displayed for the reader with hyperlinks for the years. When a year is clicked, a database is

selected for the reader based on age and the appropriate information displayed in a side frame. Readers who were children during the year in question are told which major toys were introduced during the year in question, if available. If no toys were introduced that year, then a list of toy inventions for the decade are displayed. Readers who were of high school or college age during the selected year are shown the names of major films that were released that year. And finally, readers who were adults during the year in question would see a list of major world events. Any reader who was not born or was too young to fall into the other two categories during the year in question is also shown the list of world events.

The system is surprisingly effective because it uses a detail about the reader in making a simple decision. Personal context can be provided for interval information as well. In reference to the Big Dig, access to additional information about a reader can be used to make a comparison of the duration of the construction project with what the reader personally achieved during that or a similar time interval.

Local context. As with the PLUM system, in order to provide local context, all that is needed is access to a single piece of information about readers—where they live. The drawback, however, is that in order to provide a temporal (as opposed to geographic) context, it is necessary to have access to an extensive database about happenings in local communities.

Historical context. Once the time extractor infrastructure was built, it was not necessary to restrict it to only current news articles arriving via the news wires. The Media Lab's Electronic Publishing Group,[13,23] by whom this research was conducted, maintains a database of news articles for use in various projects. The time extractor was retroactively applied to all the news in the database and by way of experiment, normalized time references in current news articles were augmented with information from other articles sharing the same time reference. In text form this information is overwhelming, but it provides an interesting overview of events across time and provides a way of providing serendipitous encounters with news articles one might not otherwise read.

Historical context can also be provided by location. Again, using knowledge of the reader's hometown, it is possible to create a view of a referenced historical period that demonstrates concurrency between events the reader is familiar with from his or her region's history and well-known or obscure historical events from other places. Other ideas include the juxtaposition of events across a variety of countries and localities that are happening concurrently, perhaps elucidating events leading up to a significant historical moment.

For example, if a reader looks at an article about Paul Revere's attempts in 1800 to refine the process of rolling copper to use as sheeting for U.S. Navy ships, the system could inform the reader that these efforts were contemporaneous with Napoleon's consolidation of power in France and the beginning of his war with Austria. Should the conclusion be drawn that the expansion into new materials of a New England area silversmith was facilitated by the U.S. government's needs to supply its ships with protective sheeting manufactured domestically because of impending difficulties in acquiring materials from overseas?

The temporal correlation between two disparate events can serve to make the original article about Paul Revere more compelling if the reader "discovers" that the event is on a direct path to intersect with a war with a profound historical significance.

An idea that was not implemented is to use either historical databases of news articles or some other database of historical writing to move a single subject matter along a time axis. In essence, this would mean to augment an article on a particular topic with information on how that topic was addressed, perceived, or understood at different points in history. In order to do this, a system must be able to determine the subject matter of both the current news article and all of the historical articles to a reasonably fine granularity. At present, such an augmentation may require more sophisticated tools for "understanding" a news article than are available. With improved understanding, historical context could be provided for the temporal intervals described in an article.

With such a system, the duration of Boston's Big Dig could be compared with other major civil engineering endeavors. Rather than complaining, with no supporting evidence, that the project is taking too long, a reader would have an opportunity to understand the scope of such projects from a historical perspective. Similarly, if the U.S. Congress is taking two months to deliberate on the budget, is this a long

Figure 2 A single news article from August 23, 1999

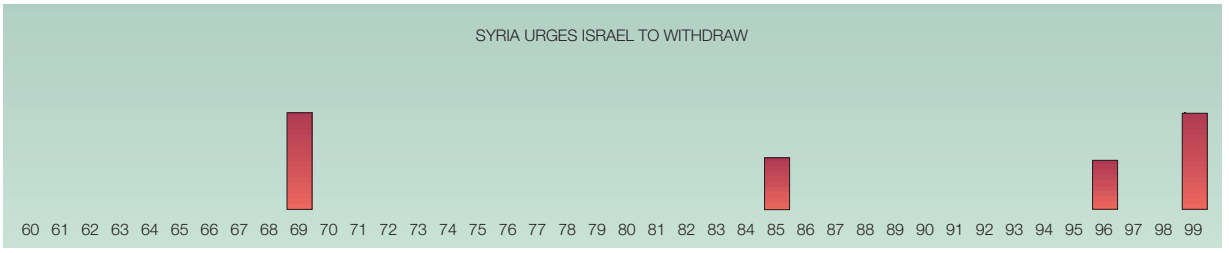


Figure 3 Tallying the temporal references from one day's news

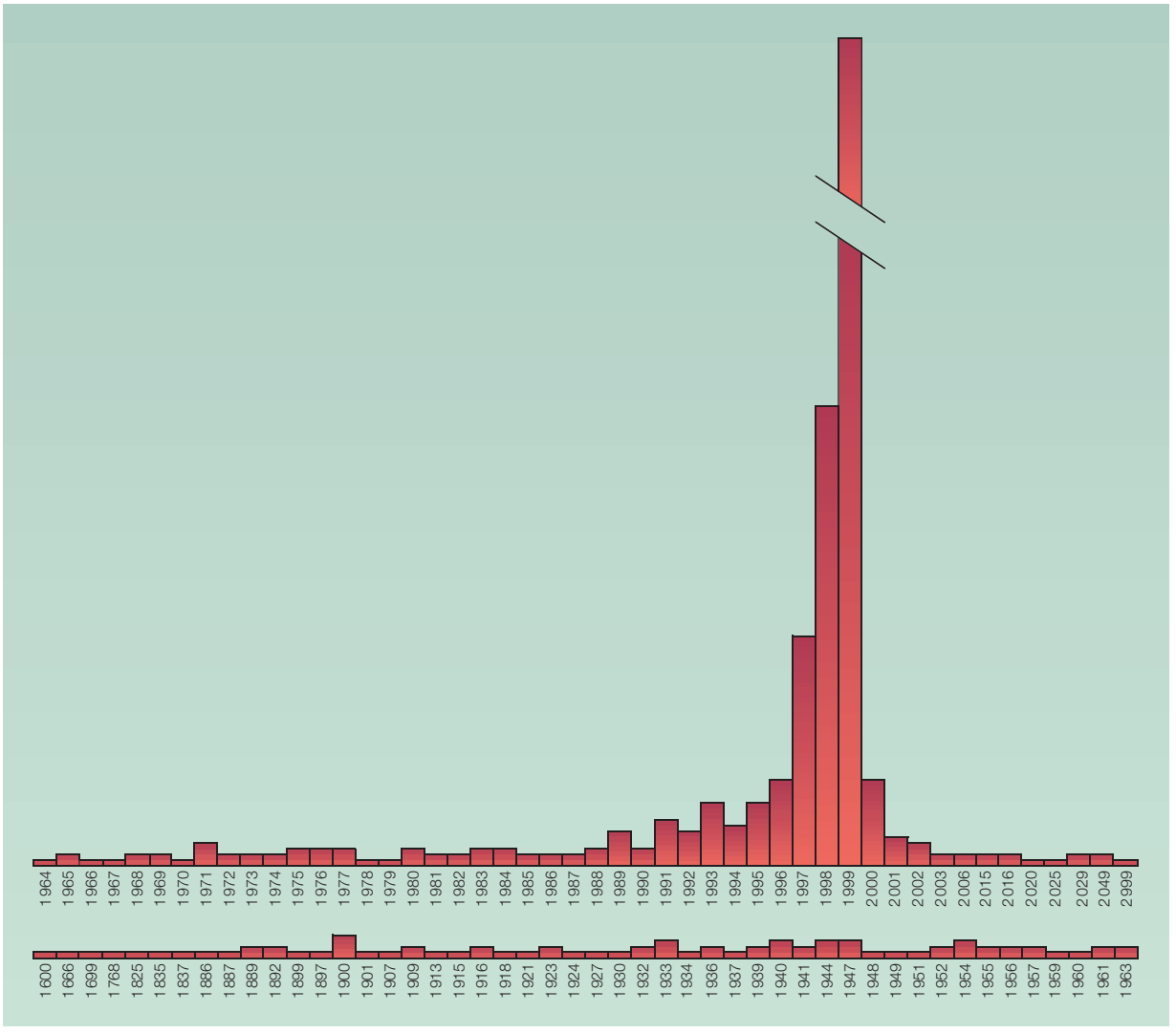
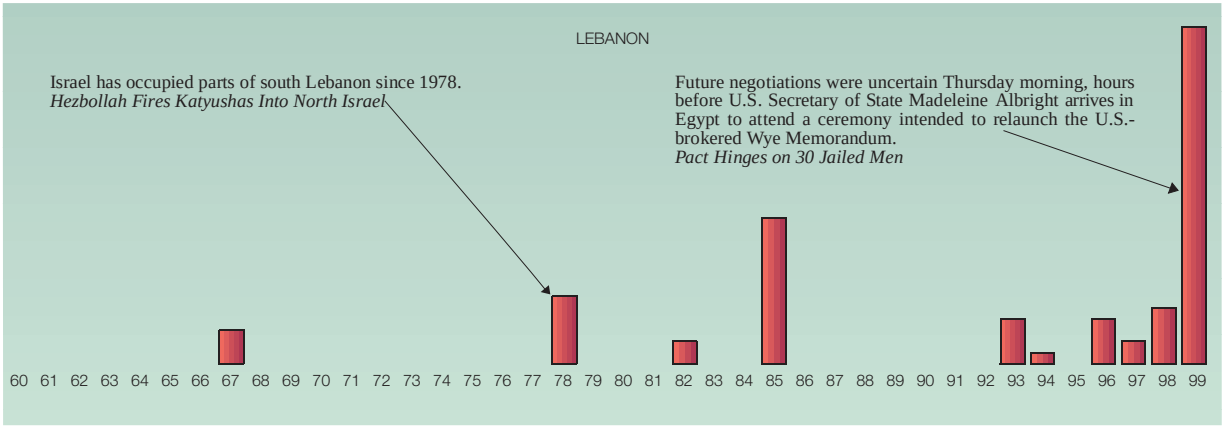


Figure 4 A time line view of a topic



time or a short time, historically speaking? How long does it take the British Parliament to debate its budget?

Visualizations. Visualizations can be used to provide readers with new ways of processing time information extracted from news articles.

Visualization of an article. A time-perspective view of a single news article allows the reader to see at a glance all of the normalizable time references in the article in a "time line" format. In this way all of the time-tagged events in the article can be perused in sequence, even if they do not appear in sequence in the article. If one of the time references in the time line is selected, the sentence from the article that generated that time reference is displayed in a small pop-up window. Selecting a block from the time line causes display of the article body at the point in the article containing the time reference. This display, shown in Figure 2, also gives the reader a sense of the span of time from the earliest reference in the article to the latest, as well as a visual understanding of the intervals between the events.

Visualization of a day's news. By enhancing a database of stored news articles, augmenting them with extracted time information, the single article time line described above has been extended to provide a time-reference view of an entire day's news. The same applet is used as in the visualization of a single article, but it is fed with data from news from an entire day and modified to accommodate moving between articles when time line blocks are clicked. In this way the reader is able to quickly see the entire span of time discussed in the news on a given day. If a particular time period fascinates the reader, he or she can quickly move among articles containing references to that time period. This interface also facilitates serendipitous discovery of articles that might not have otherwise been selected by the reader's content-selection software.

As can be seen from Figure 3, presenting all the news for an entire day in this way results in a very dense graph. The day the snapshot was taken tends to overwhelm the other time references. The year references go quite high. There may be potential for displays of this type to capture the interest of the reader, but it is likely to be of only specialized interest. For the curious, the unusual aspects of this particular example—the high year references—are due to articles with projections expressed as "in 30 years" or "in 50 years," a prediction on the ozone layer's expected progress in repairing itself, a review of a science fiction video game, and an article about the television series "Futurama," which is set in the year 2999.

Visualization of a topic. Another modification of the same basic interface, shown in Figure 4, is a collection of articles for a particular topic (selected, perhaps, by topic channels as described by Gruhl and Bender[24]) presented as an expandable time line. For the experiment, articles were collected from a search on "Lebanon" using the news service of Yahoo!** (http://news.yahoo.com). Because it is not within the scope of this research to write a system for filtering articles based on appropriateness to a searched topic,

a few irrelevant articles were removed from the data set by hand. For example, articles comparing the value of Lebanon's currency to other world currencies were removed, as were articles referring to some "Lebanon" other than the country (for example, cities and counties in the United States). Other than this filtering, all substantive articles were included; no filtering by subject matter was done.

The visualization of the Lebanon data set proved more compelling than either of the previous two uses of this visualization style. As would be expected, the greatest concentration of date references was within a day or two of the date of the search because the dates were in reference to recent events in Lebanon. The focus of other clusters of date references, however, clearly indicated the dates of significant events in Lebanon's history, as related to the current events. That is, the display does not highlight arbitrary significant events in Lebanese history, but rather historical events that are relevant to the events in Lebanon today. This is, of course, a result of the fact that date references outside of a day or two of the current date will tend to be historical references, and the more articles about a current topic that refer to a given historical event, the more prominence that event will receive in the visualization.

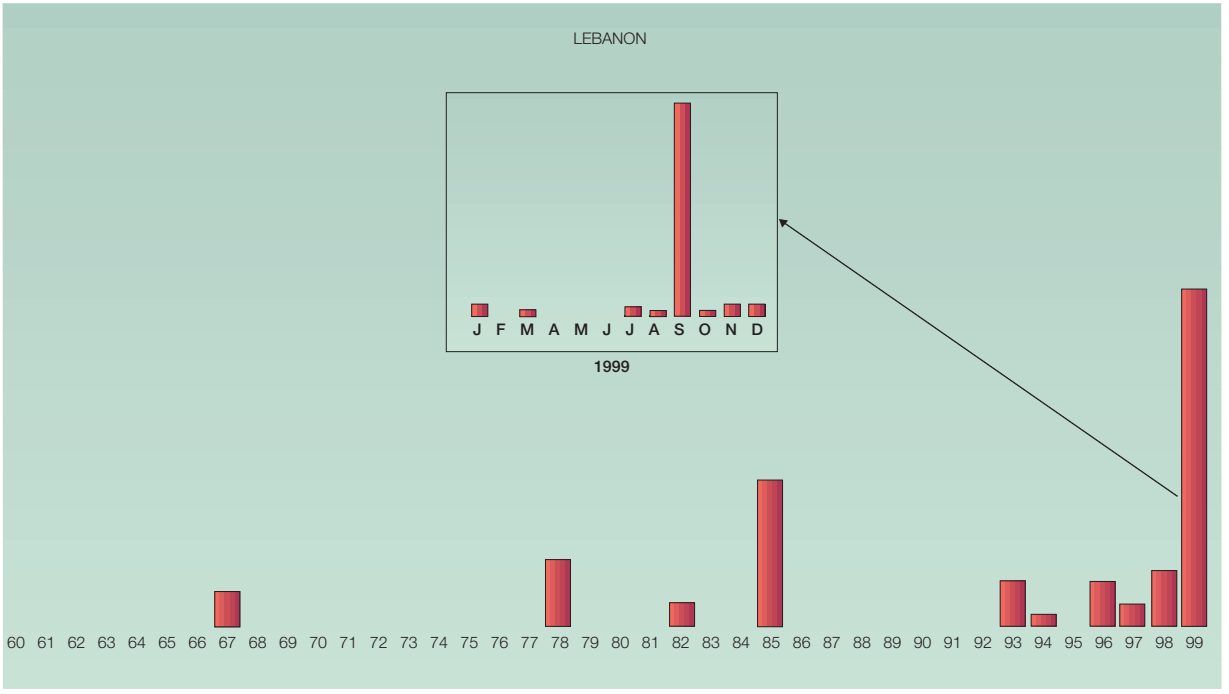
For example, the articles on this particular day were almost entirely about renewed hostilities between Lebanon and Israel. Some examples of the peaks other than at 1999 in the time line are: 1967, the date of a previous Arab-Israeli war in which Israel captured the Golan Heights region; 1978, when Israel began occupation of parts of south Lebanon; 1982, when Israel invaded Lebanon; and 1985, when Israel established a "security zone" in the frontier between the two countries. Figure 5 shows the time line expanded to show the reader additional detail for a given year. In this case the reader has expanded 1999 to show the references for each month.

Questions. This portion of Time Frames is conceptually similar to Sack's Questioning News demonstration, and is in fact based on his work. Questioning News uses natural language parsing to generate questions that might expose assumptions or implications in a news article that could bias the reader to believe something that is not explicitly stated. Our objective is to generate questions in a manner that is similar to Sack's Questioning News project; however, the intent is not to analyze the form of the sentence and expose assumptions, but rather to help the reader to ask questions about or seek additional explanation of subjects that may not be sufficiently explained in the existing text of the article. In order to do this, we first need to locate the sentence containing the time reference in question. The next step is to find familiar patterns, surrounding the reference, that will allow us to generate questions. The best approach is to locate short prepositional phrases, following the time reference, of the form "[time reference] of [object]." Then we can ask if the interval in question is adequate. Of course, it may be difficult to determine if the interval should be large or small.

An example of this type of question would be if Time Frames had found the sentence "JFK Jr. had logged 200 hours of flight time" in the news and asked, "Is 200 hours of flight time enough?" For this kind of augmentation, it is important to determine what questions make sense. It is not appropriate to generate a question for every single time reference. The interface for questions shows the sentence surrounding each time reference in the article. The ability to extract context for the time references is already in place, and looking at an article from the perspective of only these sentences proves compelling. At this stage of the augmentation process, with the time references already extracted, the grammatical parse that we were avoiding before could well prove to be valuable if applied to the sentence or paragraph immediately surrounding the references. Future work in this area might use such a parse, to determine the structure of the sentence containing the time reference, to generate meaningful questions.

Prototype interface. An interface for displaying augmentations to news should be as unobtrusive as possible. A drawback of traditional hyperlink systems is that the original page is replaced by the destination page when the hyperlink is activated. Hypertext "frames" solve this problem to some extent, but frames have other problems, most noticeably that there is no clear visual correlation between the hyperlink clicked and the frame that is filled as a result. When frames are used beyond even the simplest of implementations, the reader will often wonder just what action was performed as a result of a hyperlink jump. It is important to us that the article remain on the screen when the context information is displayed, and that the context information clearly indicate that it is not part of the article, but an extension of it. Because readers process information at different speeds, some contextual augmentations will remain valuable to them as they con-

Figure 5 Time line view of Lebanon topic with 1999 expanded



tinue to read. It should be possible for readers to keep context information available.

With these principles in mind, a prototype interface for displaying the Time Frames augmentations within a World Wide Web browser has been developed using JavaScript** and Dynamic HTML (HyperText Markup Language). Two methods of accessing the interface have been developed, but each ultimately calls a Perl package to generate an HTML document using the time extractor package. The interface is accessed either by entering an article URL to be retrieved and processed, or by typing or cutting and pasting an article body into a large text area in an HTML form. After being processed by the time extractor module, an HTML document is generated with two main DHTML layers—a main article body and a sidebar.

The entire body of the article is reproduced in the main article body layer, with time references displayed as HTML hyperlinks in the normal flow of the text. When the reader positions the cursor over one of these hyperlinks, a small DHTML layer containing contextual augmentations, colored green and called the "context" layer, slides quickly across the screen from the nearest edge. The animation serves to call the reader's attention to additional information "floating" above the body of the article. The green color indicates that the augmentation is separate from the article itself. As the reader moves the mouse away from the hyperlink, the context layer immediately vanishes. If the reader selects the hyperlink, the context layer moves across the screen to the sidebar area, where it is preserved (see Figure 6).

Integration with ZWrap. ZWrap[25] is the MIT Media Laboratory Electronic Publishing Group's current entrance into the world of publishing a newspaper to a "readership of one." A profile of each reader is stored within the system and is used to generate a newspaper tailored to that individual. This custom newspaper is then laid out and displayed in newspaper format on a computer screen. MyZWrap is built upon ZWrap, which has a structure that calls for "expert" computer programs to contribute to the knowledge about articles in the database. Time Frames is being incorporated into ZWrap as an expert on time, making MyZWrap a potential test bed

Figure 6 The prototype interface with animated augmentation

in which to continue experimentation with the Time Frames tools.

Conclusion

Time Frames extracts extensive time information from text with significant reliability, particularly within the constraints of the type of information for which it was designed—news articles. While we do not do a full natural language parse of each news article, we acknowledge that natural language parsing in conjunction with our techniques would provide vital information for improving the understanding of the time references found. After extracting the time information, the system then converts as many of these references as possible to a normalized time representation relative to the date the news article was written.

The generalization of Time Frames to a wide range of news means that the ability to generate analogies to a particular article is limited. A knowledge base that would cover all possible news topics cannot currently be created or maintained. Consequently, Time Frames attempts to provide interesting and useful augmentations to articles that do not require an understanding of the context. In the cases where additional information from the article is sought, the search is limited to the sentence surrounding the reference. A less restricted search would result in an explosion of the problem space beyond the practical scope of this research.

We approach the task of understanding news articles as a distributed and long-term process, with many "experts" in specific domains doing their part to increase the overall understanding of the text. Time Frames contributes its knowledge of time information to this resource pool. Time Frames augmentations can be used by other tools and systems.

Time Frames provides five types of augmentations to news articles afforded by the extracted time information. These augmentations are intended to improve readers' understanding of news by adding context. The augmentation classes are personal context, local context, historical context, alternative visualizations, and questions. Article augmentation is an exciting new area for the use of computation in the news dissemination process and holds the potential to help readers gain more from existing channels of news. Using the ubiquitous domain of time guarantees the applicability of these augmentations to a broad set of articles.

Acknowledgments

The authors would like to acknowledge Jon Orwant for sharing his encyclopedic knowledge of Perl and the News in the Future research consortium at MIT for sponsoring in part this work.

**Trademark or registered trademark of NeXT, Inc., The Open Group, Yahoo! Inc., or Sun Microsystems, Inc.

Cited references and note

1. N. Negroponte, *Being Digital*, Alfred A. Knopf, New York (1995).
2. P. Turetzky, *Time*, Routledge, New York and London (1998).
3. R. Levine, *A Geography of Time: The Temporal Misadventures of a Social Psychologist, or How Every Culture Keeps Time Just a Little Bit Differently*, Basic Books, New York (1997).
4. G. Dohrn-Van Rossum, *History of the Hour: Clocks and Modern Temporal Orders*, The University of Chicago Press, Chicago (1996).
5. M. Heidegger, *History of the Concept of Time*, based on lecture course at the University of Marburg in 1925, Indiana University Press, Bloomington and Indianapolis (1985).
6. J. Jesperson and J. Fitz-Randolph, *From Sundials to Atomic Clocks: Understanding Time and Frequency*, Dover Publications, Inc., New York (1977).
7. N. Dershowitz and E. M. Reingold, *Calendrical Calculations*, Cambridge University Press, Cambridge, UK (1997).
8. J. Orwant. "For Want of a Bit the User Was Lost: Cheap User Modeling," *IBM Systems Journal* **35**, Nos. 3&4, 398–416 (1996).
9. *Knowledge Discovery in Data Bases*, G. Piatetsky-Shapiro and W. J. Frawley, Editors, MIT Press, Cambridge, MA (1991).
10. G. DeJong, *Script Application: Computer Understanding of Newspaper Stories*. Ph.D. thesis, Yale University, New Haven, CT (1979).
11. A. Lippman and W. Bender, "News and Movies in the 50 Megabit Living Room," *IEEE GLOBECOM*, Tokyo (November 1987), pp. 1976–1981.
12. S. Elo, *PLUM: Contextualizing News for Communities Through Augmentation*, master's thesis, MIT Media Laboratory, Cambridge, MA (1995).
13. W. Bender, P. Chesnais, S. Elo, A. Shaw, and M. Shaw, "Enriching Communities: Harbingers of News in the Future," *IBM Systems Journal* **35**, Nos. 3&4, 369–380 (1996).
14. W. Sack, "The Questioning News System," Technical Report presented at the MIT Media Laboratory's News in the Future Consortium Meeting (May 1997).
15. R. Kullberg, *Dynamic Timelines: Visualizing Historical Information in Three Dimensions*, master's thesis, MIT Media Laboratory, Cambridge, MA (1995).
16. B. K. Smith and E. Blankinship, "Justifying Imagery: Multimedia Support for Learning Through Explanation," *IBM Systems Journal* **39**, Nos. 3&4, 749–767 (2000, this issue).
17. *Current Research in Natural Language Generation*, R. Dale, C. Mellish, and M. Zock, Editors, Academic Press, Inc., New York (1990).
18. K. Haase, "FramerD: Representing Knowledge in the Large," *IBM Systems Journal* **35**, Nos. 3&4, pp. 381–397 (1996).
19. K. Haase, "Multi-Scale Parsing Using Optimizing Finite State Machines," *Proceedings*, Association for Computational Linguistics, Columbus, OH (June 22–26, 1993).
20. D. B. Lenat and R. V. Guha, *Building Large Knowledge-Based Systems*, Addison-Wesley Publishing Co., Reading, MA (1990).
21. A. G. B. ter Meulen, *Representing Time in Natural Language: The Dynamic Interpretation of Tense and Aspect*, MIT Press, Cambridge, MA (1995).
22. Originally the date form "day/month/year" was included. However, we did not find this form in any of the news articles that were tested. It caused many false positives, so we eliminated it. We could solve the problem by making a second pass to filter out the false positives.
23. P. Chesnais, M. Mucklo, and J. Sheena, "The FishWrap Personalized News System," *Proceedings of the 2nd International Workshop on Community Networking*, Princeton, NJ (June 20–22, 1995), pp. 275–282.
24. D. Gruhl and W. Bender, "A New Structure for News Editing," *IBM Systems Journal* **39**, Nos. 3&4, 569–588 (2000, this issue).
25. D. Gruhl, *Machine Understanding of Large Bodies of Text*, Ph.D. thesis, MIT, Electrical Engineering and Computer Science, Cambridge, MA (December 1999).

Accepted for publication April 10, 2000.

Douglas B. Koen *iPhrase, 101 Rogers St., Suite 201, Cambridge, Massachusetts 02142-1049 (electronic mail: dbkoen@iphrase.com).* Mr. Koen is currently a senior software engineer at iPhrase, a company that builds products to add language interfaces to electronic commerce Web sites. He received the B.S. degree from MIT in 1995 and the M.S. degree from the MIT Media Laboratory in 2000.

Walter Bender *MIT Media Laboratory, 20 Ames Street, Cambridge, Massachusetts 02139-4307 (electronic mail: walter@media.mit.edu).* Mr. Bender is a senior scientist at the MIT Media Laboratory and principal investigator of the laboratory's News in the Future consortium. He received the B.A. degree from Harvard University in 1977 and joined the Architecture Machine Group at MIT in 1978. He received the M.S. degree from MIT in 1980. Mr. Bender is a founding member of the Media Laboratory.