Maytag: A multi-staged approach to identifying complex events in textual data

Conrad Chang, Lisa Ferro, John Gibson, Janet Hitzeman, Suzi Lubar, Justin Palmer, Sean Munson, Marc Vilain, and Benjamin Wellner
The MITRE Corporation
202 Burlington Rd.
Bedford, MA 01730 USA
contact: mbv@mitre.org (Vilain)

Abstract

We present a novel application of NLP and text mining to the analysis of financial documents. In particular, we describe an implemented prototype, Maytag, which combines information extraction and subject classification tools in an interactive exploratory framework. We present experimental results on their performance, as tailored to the financial domain, and some forward-looking extensions to the approach that enables users to specify classifications on the fly.

1 Introduction

Our goal is to support the discovery of complex events in text. By complex events, we mean events that might be structured out of multiple occurrences of other events, or that might occur over a span of time. In financial analysis, the domain that concerns us here, an example of what we mean is the problem of understanding corporate acquisition practices. To gauge a company's modus operandi in acquiring other companies, it isn't enough to know just that an acquisition occurred, but it may also be important to understand the degree to which it was debt-leveraged, or whether it was performed through reciprocal stock exchanges.

In other words, complex events are often composed of multiple facets beyond the basic event itself. One of our concerns is therefore to enable end users to access complex events through a combination of their possible facets.

Another key characteristic of rich domains like financial analysis, is that facts and events are subject to interpretation in context. To a financial analyst, it makes a difference whether a multi-million-dollar loss occurs in the context of recurring operations (a potentially chronic problem), or in the context of a one-time event, such as a merger or layoff. A second concern is thus to enable end users to interpret facts and events through automated context assessment.

The route we have taken towards this end is to model the domain of corporate finance through an interactive suite of language processing tools. Maytag, our prototype, makes the following novel contribution. Rather than trying to model complex events monolithically, we provide a range of multi-purpose information extraction and text classification methods, and allow the end user to combine these interactively. Think of it as Boolean queries where the query terms are not keywords but extracted facts, events, entities, and contextual text classifications.

2 The Maytag prototype

Figure 1, below, shows the Maytag prototype in action. In this instance, the user is browsing a particular document in the collection, the 2003 securities filings for 3M Corporation. The user has imposed a context of interpretation by selecting the "Legal matters" subject code, which causes the browser to only retrieve those portions of the document that were statistically identified as pertaining to law suits. The user has also selected retrieval based on extracted facts, in this case monetary expenses greater than $10 million. This in turn causes the browser to further restrict retrieval to those portions of the document that contain the appropriate linguistic expressions, e.g., "$73 million pre-tax charge."

As the figure shows, the granularity of these operations in our browser is that of the paragraph, which strikes a reasonable compromise between providing enough context to interpret retrieval results, but not too much. It is also ef-

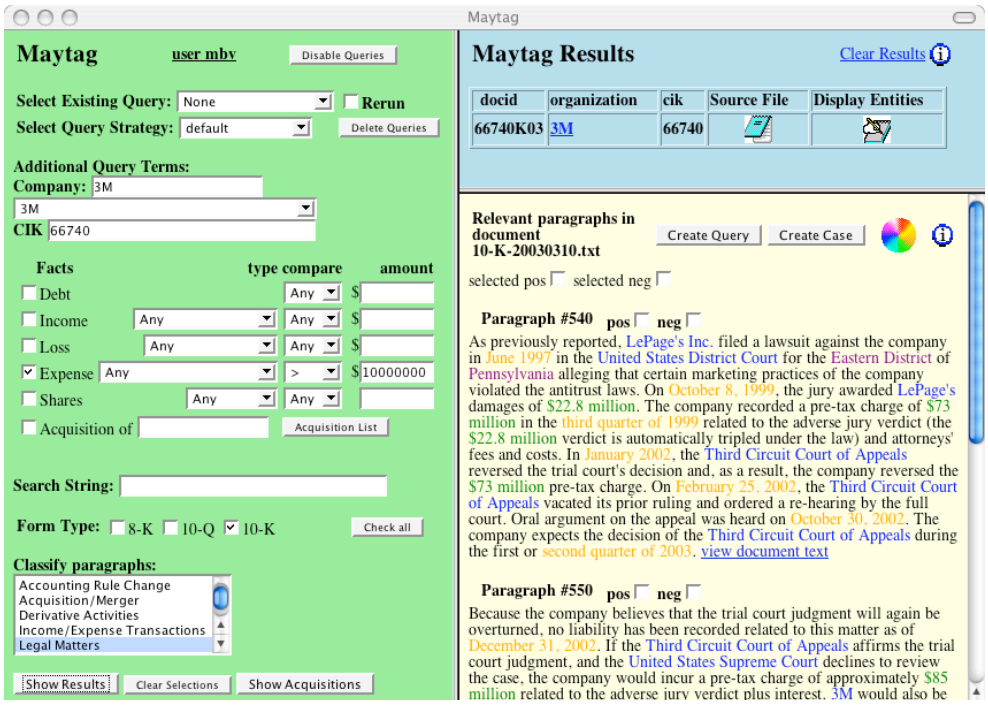

Figure 1: The Maytag interface

fective at enabling combination of query terms. Whereas the original document contains 5161 paragraphs, the number of these that were tagged with the "Legal matters" code is 27, or .5 percent of the overall document. Likewise, the query for expenses greater than $10 million restricts the return set to 26 paragraphs (.5 percent). The conjunction of both queries yields a common intersection of only 4 paragraphs, thus precisely targeting .07 percent of the overall document.

Under the hood, Maytag consists of both an on-line component and an off-line one. The on-line part is a web-based GUI that is connected to a relational database via CGI scripts (html, JavaScript, and Python). The off-line part of the system hosts the bulk of the linguistic and statistical processing that creates document meta-data: name tagging, relationship extraction, subject identification, and the like. These processes are applied to documents entering the text collection, and the results are stored as meta-data tables. The tables link the results of the off-line processing to the paragraphs in which they were found, thereby supporting the kind of extraction- and classification-based retrieval shown in Figure 1.

3 Extraction in Maytag

As is common practice, Maytag approaches extraction in stages. We begin with atomic named entities, and then detect structured entities, relationships, and events. To do so, we rely on both rule-based and statistical means.

3.1 Named entities

In Maytag, we currently extract named entities with a tried-but-true rule-based tagger based on the legacy *Alembic* system (Vilain, 1999). Although we've also developed more modern statistical methods (Burger *et* al, 1999, Wellner & Vilain, 2006), we do not currently have adequate amounts of hand-marked financial data to train these systems. We therefore found it more convenient to adapt the *Alembic* name tagger by manual hill climbing. Because this tagger was originally designed for a similar newswire task, we were able to make the port using relatively small amounts of training data. We relied on two 100+ page-long Securities filings (singly annotated), one for training, and the other for test, on which we achieve an accuracy of F=94.

We found several characteristics of our financial data to be especially challenging. The first is the widespread presence of company name look-alikes, by which we mean phrases like "Health Care Markets" or "Business Services" that may look like company names, but in fact denote business segments or the like. To circumvent this, we had to explicitly model non-names, in effect creating a business segment tagger that captures company name look-alikes and prevents them from being tagged as companies.

Another challenging characteristic of these financial reports is their length, commonly reaching hundreds of pages. This poses a quandary

for the way we handle discourse effects. As with most name taggers, we keep a "found names" list to compensate for the fact that a name may not be clearly identified throughout the entire span of the input text. This list allows the tagger to propagate a name from clear identifying contexts to non-identified occurrences elsewhere in the discourse. In newswire, this strategy boosts recall at very little cost to precision, but the sheer length of financial reports creates a disproportionate opportunity for found name lists to introduce precision errors, and then propagate them.

3.2 Structured entities, relations, and events

Another way in which financial writing differs from general news stories is the prevalence of what we've called structured entities, *i.e.*, name-like entities that have key structural attributes. The most common of these relate to money. In financial writing, one doesn't simply talk of money: one talks of a loss, gain or expense, of the business purpose associated therewith, and of the time period in which it is incurred. Consider:

> *Worldwide expenses for environmental compliance [were] $163 million in 2003.*

To capture such cases as this, we've defined a repertoire of structured entities. Fine-grained distinctions about money are encoded as *color of money* entities, with such attributes as their color (in this case, an operating expense), time stamp, and so forth. We also have structured entities for expressions of *stock shares, assets,* and *debt*. Finally, we've included a number of constructs that are more properly understood as relations (*job title*) or events (*acquisitions*).

3.3 Statistical training

Because we had no existing methods to address financial events or relations, we took this opportunity to develop a trainable approach. Recent work has begun to address relation and event extraction through trainable means, chiefly SVM classification (Zelenko *et al*, 2003, Zhou *et al*, 2005). The approach we've used here is classifier-based as well, but relies on maximum entropy modeling instead.

Most trainable approaches to event extraction are entity-anchored: given a pair of relevant entities (*e.g.*, a pair of companies), the object of the endeavor is to identify the relation that holds between them (*e.g.*, acquisition or subsidiary). We turn this around: starting with the head of the relation, we try to find the entities that fill its constituent roles. This is, unavoidably, a strongly lexicalized approach. To detect an event such as a merger or acquisition, we start from indicative head words, *e.g.*, "acquire," "purchases," "acquisition," and the like.

The process proceeds in two stages. Once we've scanned a text to find instances of our indicator heads, we classify the heads to determine whether their embedding sentence represents a valid instance of the target concept. In the case of acquisitions, this filtering stage eliminates such non-acquisitions as the use of the word "purchases" in "the company purchases raw materials." If a head passes this filter, we find the fillers of its constituent roles through a second classification stage

The role stage uses a shallow parser to chunk the sentence, and considers the nominal chunks and named entities as candidate role fillers. For acquisition events, for example, these roles include the *object* of the acquisition, the buying *agent*, the bought *assets*, the *date* of acquisition, and so forth (a total of six roles). *E.g.*

> In the *fourth quarter of 2000* (WHEN), *3M* [AGENT] also acquired *the multi-layer integrated circuit packaging line* [ASSETS] of *W.L. Gore and Associates* [OBJECT].

The maximum entropy role classifier relies on a range of feature types: the *semantic type* of the phrase (for named entities), the *phrase vocabulary*, the *distance* to the target head, and *local context* (words and phrases).

Our initial evaluation of this approach has given us encouraging first results. Based on a hand-annotated corpus of acquisition events, we've measured filtering performance at F=79, and role assignment at F=84 for the critical case of the *object* role. A more recent round of experiments has produced considerably higher performance, which we will report on later this year.

4 Subject Classification

Financial events with similar descriptions can mean different things depending on where these events appear in a document or in what context they appear. We attempt to extract this important contextual information using text classification methods. We also use text classification methods to help users to more quickly focus on an area where interesting transactions exist in an interactive environment. Specifically, we classify each paragraph in our document collection into one of several interested financial areas. Examples include: *Accounting Rule Change, Acquisitions and Mergers, Debt, Derivatives, Legal,* etc.

4.1 Experiments

In our experiments, we picked 3 corporate annual reports as the training and test document set. Paragraphs from these 3 documents, which are from 50 to 150 pages long, were annotated with the types of financial transactions they are most related to. Paragraphs that did not fall into a category of interest were classified as "other". The annotated paragraphs were divided into random 4x4 test/training splits for this test. The "other" category, due to its size, was subsampled to the size of the next-largest category.

As in the work of Nigam *et al* (2002) or Lodhi *et al* (2002), we performed a series of experiments using maximum entropy and support vector machines. Besides including the words that appeared in the paragraphs as features, we also experimented with adding named entity expressions (money, date, location, and organization), removal of stop words, and stemming. In general, each of these variations resulted in little difference compared with the baseline features consisting of only the words in the paragraphs. Overall results ranged from F-measures of 70-75 for more frequent categories down to above 30-40 for categories appearing less frequently.

4.2 Online Learning

We have embedded our text classification method into an online learning framework that allows users to select text segments, specify categories for those segments and subsequently receive automatically classified paragraphs similar to those already identified. The highest confidence paragraphs, as determined by the classifier, are presented to the user for verification and possible re-classification.

Figure 1, at the start of this paper, shows the way this is implemented in the Maytag interface. Checkboxes labeled *pos* and *neg* are provided next to each displayed paragraph: by selecting one or the other of these checkboxes, users indicate whether the paragraph is to be treated as a positive or a negative example of the category they are elaborating. In our preliminary studies, we were able to achieve the peak performance (the highest *F1 score*) within the first 20 training examples using 4 different categories.

5 Discussion and future work

The ability to combine a range of analytic processing tools, and the ability to explore their results interactively are the backbone of our approach. In this paper, we've covered the framework of our Maytag prototype, and have looked under its hood at our extraction and classification methods, especially as they apply to financial texts. Much new work is in the offing.

Many experiments are in progress now to assess performance on other text types (financial news), and to pin down performance on a wider range of events, relations, and structured entities.

Another question we would like to address is how best to manage the interaction between classification and extraction: a mutual feedback process may well exist here.

We are also concerned with supporting financial analysis across multiple documents. This has implications in the area of cross-document coreference, and is also leading us to investigate visual ways to define queries that go beyond the paragraph and span many texts over many years.

Finally, we are hoping to conduct user studies to validate our fundamental assumption. Indeed, this work presupposes that interactive application of multi-purpose classification and extraction techniques can model complex events as well as monolithic extraction tools *à la* MUC.

Acknowledgements

This research was performed under a MITRE Corporation sponsored research project.

References

Zhou, G., Su J., Zhang, J., and Zhang, M. 2005. Exploring various knowledge in relation extraction. *Proc. of the 43rd ACL Conf*, Ann Arbor, MI.

Nigam, K., Lafferty, J., and McCallum, A. 1999. Using maximum entropy for text classification. *Proc. of IJCAI '99 Workshop on Information Filtering*.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, and N., Watkins, C. 2002. Text classification using string kernels. *Journal of Machine Learning Research*, Vol. 2, pp. 419-444.

Vilain, M. and Day, D. 1996. Finite-state Phrase Parsing by Rule Sequences, *Proc. of COLING-96*.

Vilain, M. 1999. Inferential information extraction. In Pazienza, M.T. & Basili, R., *Information Extraction*. Springer Verlag.

Wellner, B., and Vilain, M. (2006) Leveraging machine readable dictionaries in discriminative sequence models. *Proc. of LREC 2006* (to appear).

Zelenko D., Aone C. and Richardella. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research*. pp1083-1106.