Mining Measured Information from Text

Arun S. Maiya, Dale Visser, and Andrew Wan
Institute for Defense Analyses — Alexandria, VA, USA
{amaiya, dvisser, awan}@ida.org

ABSTRACT

We present an approach to extract *measured information* from text (*e.g.,* a 1370 $^\circ C$ melting point, a BMI greater than 29.9 kg/m^2). Such extractions are critically important across a wide range of domains — especially those involving search and exploration of scientific and technical documents. We first propose a rule-based entity extractor to mine *measured quantities* (*i.e.,* a numeric value paired with a measurement unit), which supports a vast and comprehensive set of both common and obscure measurement units. Our method is highly robust and can correctly recover valid measured quantities even when significant errors are introduced through the process of converting document formats like PDF to plain text. Next, we describe an approach to extracting the *properties* being measured (*e.g.,* the property *pixel pitch* in the phrase "a pixel pitch as high as 352 μm"). Finally, we present MQSearch: the realization of a search engine with full support for *measured information.*

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text Analysis*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Search Process*

Keywords

text mining, information retrieval, information extraction, measured quantities, numerical queries

1. INTRODUCTION AND MOTIVATION

Scientific and technical documents describe methods and results using *measured quantities*: a numeric value paired with a unit of measurement. Examples of text snippets containing such measured quantities include:

- *average gravity curvature* $\zeta = (1.3999 \pm 0.003) \times 10^{-5} s^{-2} m^{-1}$
- 12 $^\circ C$ *melting point*
- *distance from Earth to the Sun is* 9.3×10^7 *miles*
- *average responsivity as low as* 6.2 pA/K

Note that these measured quantities (*e.g.,* 6.2 pA/K) are typically associated with a specific *measured property* (*e.g.,* average responsivity). In this paper, we study ways in which to extract these kinds of *measured information* from documents.[1] The mining of such information is critically important across many domains — especially those involving search and exploration of scientific and technical articles. For instance, an optics researcher may wish to know if the performance of Nd:YAG laser-pumped KTP parametric oscillators has ever been tested at wavelengths longer than 2.4 μm. Full-text search engines using inverted indexes allow ad hoc queries on terms such as "KTP parametric oscillator", but the ability to further filter search results based on wavelengths greater than 2.4 μm is *not* typically supported. To accomplish this, one must first identify and extract valid *measured quantities* (*e.g.,* 2.4 μm) in unstructured text and, then, identify and extract the *properties* being measured (*e.g., wavelength*). These extractions could then be stored in the index of a search engine in a way that supports subsequent document queries on measured information (*e.g.,* faceted navigation, numeric range queries).

Surprisingly, there is very little existing work on how best to realize this process. Lines of research most closely related to the present work include extracting numerical attributes (*e.g.,* [1,4]), supporting numerical document queries (*e.g.,* [5,12]), and formula identification (*e.g.,* [7]). However, none of these existing works address the comprehensive extraction of and search for measured information in document data, as described above. Indeed, numerous challenges exist in such scenarios. Many widely-used, full-text search engines (*e.g.,* Apache Solr) convert the original document format to *plain text* prior to indexing and storage — an *extremely* error-ridden process. For instance, in the extracted text, exponents are typically lost (*e.g.,* 10^5 becomes 105, s^{-2} becomes s–2). Moreover, the conversion of some characters can be highly inconsistent and unpredictable. A simple minus sign can be converted to a range of different dash characters or even "garbage" characters. The same is true for other symbols such as μ, multiplication signs, and degree symbols. It is this inconsistent and error-ridden text, then, that is ultimately stored in the index of the search engine making it virtually impossible to adequately locate documents by measured quantities. Without the correct identification of measured quantities, it is virtually impossible to identify *properties* being measured, which are critical in efficiently

SIGIR '15, August 9–13, 2015, Santiago, Chile.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-3621-5/15/08...$15.00
DOI: http://dx.doi.org/10.1145/2766462.2767789.

[1] We define *measured information* as *measured quantities* and the *measured properties* to which they are associated.

navigating scientific and technical articles for state-of-the-art information. In general, there is a great deal of heterogeneity in how *measured quantities* and *measured properties* appear in text – both naturally and through corruption. This, then, motivates the current investigation of how best to extract such information.

Recent studies [3, 6] have revealed that rule-based approaches to information extraction tend to be more effective, interpretable, and customizable than state-of-the-art machine learning approaches. We employ rule-based extraction methods in this work. Our contributions are as follows:

- We propose and describe a rule-based entity extractor to identify *measured quantities* in unstructured text documents. Our method includes an error-correcting procedure that recovers from aforementioned text conversion errors by 1) *reverse engineering* the corrupted and mangled measured quantities back to their original, correct form and 2) *standardizing* this form for storage in an inverted index and subsequent query processing.
- Using these extracted measured quantities, we show how to further extract the *measured properties* to which they are associated.
- Finally, we present MQSearch: the realization of a search engine with full support for *measured information.* MQSearch is a facet-based navigation system that allows users to navigate large document sets based on measured quantities, measured properties, and the topics and themes to which they are associated. To the best of our knowledge, no other search engine in existence fully supports such a capability.

We begin with describing the extraction of *measured quantities.*

2. MEASURED QUANTITIES

We view *measured quantities* as a 5-tuple of the form: (*sign*, *<u>number</u>*, *error*, *scientific notation*, *<u>units</u>*), where underlined elements are mandatory and others are optional. As an example, a team of researchers in Italy recently reported the first direct measurement of gravity's curvature as $(1.3999 \pm 0.003) \times 10^{-5} s^{-2} m^{-1}$ [10]. The corresponding 5-tuple representation of this[2] is:

(<empty>, 1.3999, 0.003, 10^{-5}, $s^{-2}m^{-1}$).

5-tuples such as this are populated using a series of extraction rules that operate on individual sentences. These rules fall into four broad categories: 1) pre-processing, 2) units, 3) quantities, and 4) post-processing. Simplified forms of some of the rules for units and quantities are shown in Table 1.[3] We refer to the algorithm implementing such rules as *Measured Quantity Extractor* or **MQE**. We begin with pre-processing rules.

Pre-Processing. As mentioned previously, when extracting text from various document formats (*e.g.,* PDF, MS Office), characters often appear inconsistently. Minus signs, multiplication signs (*e.g.,* ×, ·), equal-like symbols (*e.g.,* ≈, ≃, ≅), degree symbols, and the μ character can appear in a variety of ways or, in some cases, as "garbage" characters. For instance, minus can appear as the *en dash* character or appear corrupted as â€. Pre-processing rules identify these variations in text and perform the necessary normalization for accurate extraction of units and quantities.

Units. A *measurement unit* preceded by a numeric string conforming to the 5-tuple structure is the base indicator of a *measured quantity.* Thus, to identify valid *measured quantities*, we require a comprehensive ontology of units. We obtained an initial units ontology from the OBO Foundry,[4] but this was quite incomplete. We, then, expanded the ontology using largely public sources (*e.g.,* `convert-me.com`, DoD technical reports, Physical Review, Nature Communications). Each unit has an associated rule. An example rule for *m* (*i.e.,* symbol for meters) is shown in Rule 5 of Table 1. Note that such rules include optional prefixes for submultiples and multiples (*e.g.,* μ before *m*, *kilo* before *meter*). Unit rules, when combined with pre-processing rules described previously, can accurately extract units under a range of noisy conditions. For instance, the corrupted unit *m*â€1 is correctly recovered as m^{-1} by MQE. Finally, as shown in Rules 6 and 7, compound units are also supported (*e.g.,* km/h, kilometer per hour, $s^{-2} \cdot m^{-1}$).

Quantities. Like units, quantities (*i.e.,* numbers with optional error ranges and scientific notation) can appear in a range of ways due to both corruptions and natural variation. These variations are collectively captured by rules such as those shown in Table 1 (*i.e.,* Rules 1-4), which populate the remainder of the 5-tuple structure. As shown in Table 1, such rules capture a wide range of quantity formats (*e.g.,* $10,000$ with a comma, $1.3999 \pm 0.003 \times 10^{-5}$ with both an error range and scientific notation, 1.23×105 with lost exponent in 10^5). To support numeric range queries, extracted quantities are standardized prior to storage in a search engine index (*e.g.,* the extracted quantity $1.3999 \pm 0.003 \times 10^{-5}$ is stored simply as 0.000013999) [11].

Post-Processing. We have already seen that text extracted from various document formats can be noisy. For instance, information from tables, headers, and figures can sometimes result in seemingly random sequences of numbers and letters in extracted text. In some cases, such information can erroneously be picked up by aforementioned rules as *measured quantities.* This is especially true for single letter units (*e.g., m* for meters, *A* for Ampere, etc.). Post-processing rules are employed to reject such extractions and minimize false positives. Examples of such rejection rules include context-based rules (*e.g.,* reject when preceded by "Table" or "Figure"), repetition-based rules such as rejecting compound units consisting of repeated single letter units (*e.g.,* 3 AJmm), and allowing a dash only between certain quantities and units (*e.g., 10-cm* is okay but not *10-A*).

As we will show in Section 4, when used in combination, these rules collectively enable highly accurate extractions of *measured quantities* – which, in turn, can be exploited to extract the *properties* being measured, as described next.

[2] Since there is no explicit sign in this example, the first element is left empty.

[3] Rules are shown in Perl-like syntax, the de facto standard for regular expressions.

[4] `http://www.obofoundry.org/`

Rule	Pattern	Example Matches
1) number	[+−]?(\d((\d?\d?[,]\d{2,3}([,]\d{2,3})*)|\d*))(\.(\d[\d\s]*\d|\d))?	1000.05, +5, -0.2, and 1,000
2) number (leading point)	[+-]?\.\d(\d\d(\s\d{3})+(\s\d{1,3})?|\d*)	-.98, .04, +.755
3) error	(\s{0,2} ± \s{0,2}[\d.]+)?	±0.003 in "1.3999 ± 0.003"
4) sci. notation.	(\s*[eE]|\s*([xX×])\s*10 *\^? [+-]?\d+)?	*e.g.,* forms of $\times 10^{-5}$: ×105, e-5, E-5
5) unit	*e.g.,* [fpnμmcdk]?m([\^]?[2-6] | [\-][1-6]) — $m^{\#}$ **normalized to** $m\hat{}\#$	μm, m–1 (m^{-1}), $cm2$ (cm^2), $cm\hat{}2$
6) connector	(\s?/\s? | [Pp]er |-per-| [-\s× ·*])?	per, /, ·, ×
7) compound unit	<unit>(<connector><unit>)+	km/h, kilometer per hour,km·h^{-1}

Table 1: **[MQE Rules.]** Simplified forms of some rules for extraction of *measured quantities*.

Pattern	Example Matches (two examples shown for each rule)
NP SYM{0,2} EQ mq	1) *gravity curvature* $\zeta = 1.4 \times 10^{-5} s^{-2} m^{-1}$ 2) *floor area* ≊ $32m^2$
mq IN? NP	1) *a* 352 *μm pixel pitch* 2) 50*mL of 30% fuming sulfuric acid*
NP IN DT? NP VP+ (TO|IN|RB|JJ)* mq	1) *strength of panel was set to 9 ksi* 2) *freq. of scans was roughly 300 Hz*
NP (IN DT? NP)* VP+ (IN|TO|RB|JJ)* mq	1) *pixel pitch employed was* 352 *μm.* 2) *panel strength was recorded at* 9 *ksi.*
NP (CC|IN|TO|RB|JJ)* \(?mq\)?	1) *wavelengths of at least* 2.4 *μm* 2) *panel strength (*9 *ksi)*

Table 2: **[MPE Rules.]** Simplified forms of some syntactic patterns to extract *measured properties*.

3. MEASURED PROPERTIES

We now turn our attention to the extraction of *measured properties*. To better illustrate the problem, we show several example snippets containing measured quantities. In each example, the *measured quantity* is shown in blue, the *property* being measured is highlighted in red, and the characters connecting them are underlined:

- *a pixel pitch as high as roughly* 352 *μm*
- *a* 352 *μm pixel pitch*
- *The pixel pitch employed was* 352 *μm.*
- *average gravity curvature* $\zeta =(1.3999 \pm 0.003) \times 10^{-5} s^{-2} m^{-1}$
- *with* 50*mL of 30% fuming sulfuric acid*
- *size* ≅ $0.1m^2$
- *frequency of longitudinal scan was approximately 300 Hz.*
- *a nominal current density of 1.3 A/cm*^{2} *to 0.03 A/cm*^{2}
- *panel strength lower than 8.90 ksi (61.4 MPa)*
- *wavelengths at least* 2.4 *μm*
- *large fields of about, or above 10 kV/cm*

From just the examples shown, it is easy to see that there is an extremely high degree of variability in the words connecting a *measured property* with a *measured quantity*. These examples represent just a small sample of the many possible variations. However, upon closer inspection, we find that this variability can be reduced to a small number of syntactic patterns based on part-of-speech (POS) that capture most scenarios. Table 2 shows some syntactic patterns that we employ to extract *measured properties*. We refer to the extractor applying such syntactic rules as *Measured Property Extractor* or **MPE**.

In Table 2, noun phrases shown in red (*i.e.,* NP) are extracted and taken as the *measured property*. Measured quantities are represented in blue by mq. The EQ tag represents all symbols related to '=' (*e.g.,* ≈, ≃). The SYM tag matches one or two character symbols (*e.g.,* a greek letter). Other symbols (*e.g.,* JJ, RB, IN, CC, VP) are part-of-speech tags in Penn Treebank format. Note that tags such as RB (*i.e.,* an adverb) should be taken to include variations such as the comparative and superlative forms. This is not explicitly shown for reasons of brevity. This small set of patterns matches a very wide range of possible phrase combinations for *measured properties* and are executed sequentially in the order shown. We implemented MPE using the Brill part-of-speech tagger [2]. As we will show in the next section, the accuracy with which our algorithms are able to extract measured properties and measured quantities is remarkable — especially given the aforementioned issues with noisy and corrupted input text.

4. EXPERIMENTAL EVALUATION

Since our research is sponsored by the U.S. Department of Defense (DoD), we evaluate our approach on a text corpus consisting of 40,807 unclassified research reports published in the 2008-2010 time frame and hosted by the Defense Technical Information Center (DTIC). This rich collection describes a wide range of research funded by the DoD spanning numerous fields from engineering and physical science to biomedical research and social science. The DTIC documents considered in this paper have been approved for public release and unlimited distribution. All documents are in PDF format, and text was extracted from them using the `pdftotext` utility.[5] From this collection, we generated samples using the following procedure. To evaluate the ability of MQE to extract measured quantities, we sampled uniform random sentences from the population of all sentences containing a numeric value. By examining sentences with a number (but not necessarily a measurement unit), we are able to accurately identify false negatives in addition to false positives. Next, to evaluate the ability of MPE to extract measured properties, we generated a random sample of sentences from the population of all sentences containing a measured quantity, as identified by MQE. We employed sample sizes of 1000 and 500 for MQE and MPE, respectively. This produced sufficient 95% confidence bounds on our estimates for precision and recall over the entire corpus. Different fields employ different measures in different ways. By considering sentences sampled randomly in this fashion, we are able to evaluate our methods on text data that capture the diverse ways in which measured information is reported across different fields. To the best of our knowledge, no other approaches exist for extracting such *measured information* from scientific and technical documents. Thus, there are no appropriate baselines against which our meth-

[5] `http://www.foolabs.com/xpdf/home.html`

ods can be compared. Table 3 shows the precision and recall estimates for both the measured quantity extractor and the measured property extractor over the entire corpus.

Extractor	Precision	Recall
MQE	(0.93, 0.99)	(0.92, 0.99)
MPE	(0.93, 0.97)	(0.88, 0.94)

Table 3: **95% Confidence Intervals** for precision and recall when extracting *measured quantities* (using MQE) and *measured properties* (using MPE) from the DTIC corpus.

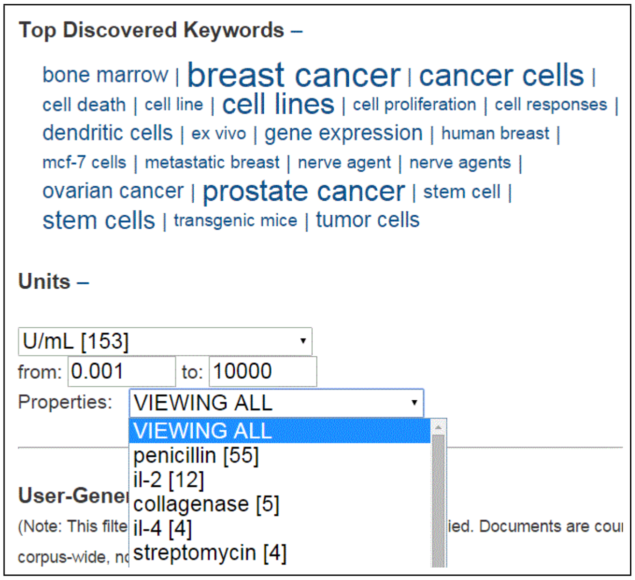
As can be seen in the table, both MQE and MPE perform extraordinarily well in extracting *measured quantities* and the *properties* they describe from documents across disparate fields. Having demonstrated the success with which *measured information* can be mined, we now demonstrate how these extractions can be exploited in novel search applications.

5. AN APPLICATION: MQSEARCH

Here, we present MQSEARCH: a realization of a search engine with full support for *measured information*. MQSEARCH is implemented using Apache Solr[6] and AJAX Solr[7], both of which support full-text search, faceted navigation, and numeric range queries. During the process of indexing and ingesting the DTIC document set into our search engine, we apply our extractors to encountered text and store both *measured quantities* and *measured properties* in the search engine index. In addition, the search engine performs keyphrase extraction on documents using the KERA algorithm described in [8]. Using Solr filter queries, extracted keyphrases can be used to produce a tag cloud for any subset of the document set. Figure 1 shows the faceted navigation panel of MQSEARCH, which allows users to filter documents based on discovered measurement units, quantity ranges, and measured properties. In Figure 1, the measurement unit *U/mL* is selected. We see that there are 153 documents (out of roughly 40,000) mentioning this unit with quantities ranging from 0.001 U/mL to 10,000 U/mL. The property most frequently measured in *U/mL* is *penicillin*. From the tag cloud, we see that documents containing quantities measured in *U/mL* tend to cover topics such as breast cancer and prostate cancer research.[8] The search results can be filtered further along any of these dimensions. Filtering by LDA-discovered topics is also supported but not shown in the figure [9]. To the best of our knowledge, ours is the first search engine with such support for *measured information*.

6. CONCLUSION

In this paper, we have proposed a demonstrably effective approach to extracting *measured information* from unstructured text data. We showed both how to extract *measured quantities* and the *properties* being measured. We further demonstrated how such extractions might be used in a search engine for documents rich in measured information. To the best of our knowledge, no other search engine in existence supports such functionality. Our extraction methods have the potential to substantially improve search, navigation, and exploratory analysis of large or even massive collections of scientific and technical articles. For future work, we plan on marrying our proposed approaches with other well-studied techniques for exploratory search.



Top Discovered Keywords –

bone marrow | breast cancer | cancer cells | cell death | cell line | cell lines | cell proliferation | cell responses | dendritic cells | ex vivo | gene expression | human breast | mcf-7 cells | metastatic breast | nerve agent | nerve agents | ovarian cancer | prostate cancer | stem cell | stem cells | transgenic mice | tumor cells

Units –

U/mL [153]

from: 0.001 to: 10000

Properties: VIEWING ALL

VIEWING ALL
penicillin [55]
il-2 [12]
collagenase [5]
il-4 [4]
streptomycin [4]

User-Gene

(Note: This filte ied. Documents are cou

corpus-wide, n

Figure 1: [MQSEARCH.] The measurement unit *U/mL* is selected, which reveals the associated topics (*e.g.,* breast/prostate cancer), associated measured properties (*e.g.,* concentrations of *penicillin*), and associated quantity ranges (*i.e., 0.001* to *10,000*).

7. REFERENCES

[1] A. Bakalov, A. Fuxman, P. P. Talukdar, and S. Chakrabarti. Scad: collective discovery of attribute values. In *WWW '11*.

[2] E. Brill. A Simple Rule-based Part of Speech Tagger. In *ANLC '92*.

[3] L. Chiticariu, Y. Li, and F. R. Reiss. Rule-based Information Extraction is Dead! Long Live Rule-based Information Extraction Systems! In *EMNLP '13*.

[4] D. Davidov and A. Rappoport. Extraction and Approximation of Numerical Attributes from the Web. In *ACL '10*.

[5] M. Fontoura, R. Lempel, R. Qi, and J. Zien. Inverted Index Support for Numeric Search. *Internet Math.*, 3(2):153–186, 2006.

[6] S. Gupta and C. Manning. SPIED: Stanford Pattern based Information Extraction and Diagnostics. In *Proc. 2014 Workshop on Interactive Language Learning, Visualization, and Interfaces*.

[7] X. Lin, L. Gao, Z. Tang, X. Lin, and X. Hu. Mathematical Formula Identification in PDF Documents. In *ICDAR '11*.

[8] A. S. Maiya, J. P. Thompson, F. L. Lemos, and R. M. Rolfe. Exploratory Analysis of Highly Heterogeneous Document Collections. In *KDD '13*.

[9] A. K. McCallum. MALLET: A Machine Learning for Language Toolkit, 2002.

[10] G. Rosi, L. Cacciapuoti, F. Sorrentino, M. Menchetti, M. Prevedelli, and G. M. Tino. Measurement of the Gravity-Field Curvature by Atom Interferometry. *Physical Review Letters*, 114(1), Jan. 2015.

[11] U. Schindler and M. Diepenbroek. Generic XML-based Framework for Metadata Portals. *Comput. Geosci.*, 34(12):1947–1955, Dec. 2008.

[12] H. Seidl, F. I. Informatik, T. Schwentick, and A. Muscholl. Numerical Document Queries. In *PODS 2003*. ACM, June 2003.

[6] `http://lucene.apache.org/solr/`

[7] `https://github.com/evolvingweb/ajax-solr`

[8] This cancer research was funded by U.S. Army MRMC through a congressionally directed research program.