Bootstrapping a historical commodities lexicon with SKOS and DBpedia

Ewan Klein
ILCC, School of Informatics
University of Edinburgh
EH8 9AB, Edinburgh, UK
ewan@inf.ed.ac.uk

Beatrice Alex
ILCC, School of Informatics
University of Edinburgh
EH8 9AB, Edinburgh, UK
balex@inf.ed.ac.uk

Jim Clifford
Department of History
University of Saskatchewan
Saskatoon, SK S7N 5A5, Canada
jim.clifford@usask.ca

Abstract

Named entity recognition for novel domains can be challenging in the absence of suitable training materials for machine-learning or lexicons and gazetteers for term look-up. We describe an approach that starts from a small, manually created word list of commodities traded in the nineteenth century, and then uses semantic web techniques to augment the list by an order of magnitude, drawing on data stored in DBpedia. This work was conducted during the *Trading Consequences* project on text mining and visualisation of historical documents for the study of global trading in the British empire.

1 Introduction

The *Trading Consequences* project[1] aims to assist environmental historians in understanding the economic and environmental consequences of commodity trading during the nineteenth century. We are applying text mining to large quantities of historical text in order to convert unstructured textual information into structured data that can be queried and visualised. While prior historical research into commodity flows (Cronon, 1991; Cushman, 2013; Innis and Drache, 1995; McCook, 2006; Tully, 2009) has focused on a small number of widely traded natural resources, the large corpora of digitised documents processed by *Trading Consequences* is giving historians data about a much broader range of commodities. A detailed appraisal of trade in these resources will yield a significantly more accurate picture of globalisation and its environmental consequences.

In this paper we focus on our approach to building a lexicon to support the recognition of commodity terms in text. We provide some background to this work in Section 2. In Section 3, we describe the process of creating the lexicon; this starts from a manually collected seed set of commodity terms which is then expanded semi-automatically using DBpedia.[2] An evaluation of the quality of the commodity lexicon is provided in Section 4.

2 Background

Figure 1 shows an overview of the architecture of the *Trading Consequences* system. Input documents are processed by the text mining pipeline, which is based on the LT-XML2[3] and LT-TTT2[4] toolkits (Grover et al., 2008). After initial format conversion, the text undergoes language identification and OCR post-correction and normalisation.[5] It is then processed further by shallow linguistic analysis, lexicon and gazetteer lookup, named entity recognition and grounding, and relation extraction (see Figure 2).

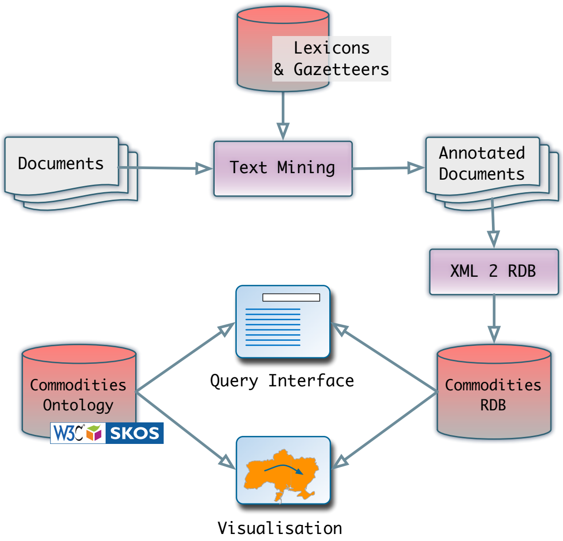


Figure 1: Architecture of the *Trading Consequences* prototype.

In *Trading Consequences*, we determine which commodities were mentioned when and in relation to which

[1] http://tradingconsequences.blogs.edina.ac.uk/

[2] http://www.dbpedia.org

[3] LT-XML2 includes APIs for parsing XML documents (both as event streams and as trees), creating them, serialising them and navigating them with XPath queries; see http://www.ltg.ed.ac.uk/software/ltxml2.

[4] LT-TTT2 is built around the LT-XML2 programs and provides NLP components for a variety of text processing tasks such as tokenisation and sentence-splitting, chunking and rule-based named entity recognition. It includes a third party part-of-speech tagger and lemmatiser; see http://www.ltg.ed.ac.uk/software/lt-ttt2.

[5] For more details on dealing with OCR errors, see (Lopresti, 2008; Alex et al., 2012).

Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH) @ EACL 2014, pages 13–21,
Gothenburg, Sweden, April 26 2014.

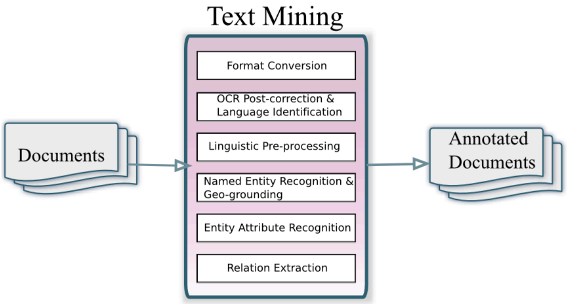

Text Mining
Format Conversion
OCR Post-correction & Language Identification
Linguistic Pre-processing
Named Entity Recognition & Geo-grounding
Entity Attribute Recognition
Relation Extraction
Documents
Annotated Documents

Figure 2: Architecture of the text mining component

locations. We also determine whether locations are mentioned as points of origin, transit or destination and whether vocabulary relating to diseases and disasters appears in the text. All mined information is added back into the XML documents as different layers of stand-off annotation.

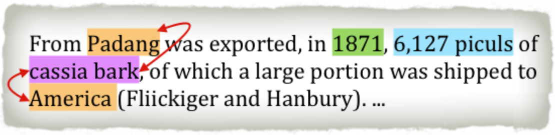
The annotations are subsequently used to populate a relational database. This stores not just metadata about the individual document, but also detailed information that results from the text mining, such as named entities, relations, and how these are expressed in the relevant document in context. Visualisations and a query interface access the database so that users can either search the mined information directly through textual queries or browse the data in a more exploratory manner. A temporal dimension for the visualisation is provided by correlating commodity mentions in documents with the publication date of those documents. All information mined from the collections is linked back to the original documents of the data providers.

We analyse textual data from a variety of sources, including the House of Commons Parliamentary Papers (HCPP)[6] from ProQuest;[7] the Early Canadiana Online data archive (ECO) from Canadian.org;[8] the Directors' Correspondence Collection from the Archives at Kew Gardens available at Jstor Global Plants (LETTERS);[9] Adam Matthew's Confidential Print collections (CPRINT);[10] and a subpart of the Foreign and Commonwealth Office Collection (FCOC) from Jstor.[11] Together these sources amount to over 10 million pages of text and over 7 billion word tokens. Table 1 provides an overview of the number of documents and OCR scan images per collection or sub-collection available to the *Trading Consequences* consortium.

We used a variety of techniques for carrying out named entity recognition, covering not only commodities, but also places, dates and amounts. Figure 3 shows some of the entities which we extract from the text, e.g. the places *Padang* and *America*, the year *1871*, the commodity *cassia bark* and the quantity and unit *6,127 piculs*. We are also able to identify that *Padang* is an origin location and *America* is a destination location and to ground both locations to geographical coordinates. The commodity-place relations *LOC(cassia bark, Padang)* and *LOC(cassia bark, America)*, visualised by the red arrows in Figure 3, are also identified. In this paper, our focus is on commodity mentions, and we will discuss these in more detail in the next section.

Collection	# of docs	# of images
HCPP	118,526	6,448,739
ECO	83,016	3,938,758
LETTERS	14,340	n/a
CPRINT	1,315	140,010
FCOC	1,000	41,611

Table 1: Number of documents and images per collection. One image usually corresponds to one document page, except in the case of CPRINT, where it mostly corresponds to two document pages. The LETTERS collection does not contain OCRed text but summaries of hand-written letters.



From Padang was exported, in 1871, 6,127 piculs of cassia bark, of which a large portion was shipped to America (Flückiger and Hanbury). ...

Figure 3: Excerpt from *Spices* (Ridley, 1912). Extracted entities are highlighted in colour and relations are visualised using arrows.

3 Lexicon Construction

In recent years, the dominant paradigm for NER has been supervised machine learning (Tjong Kim Sang and De Meulder, 2003). However, to be effective, this requires a considerable investment of effort in manually preparing suitable training data. Since we lacked the resources to create such data, we decided instead to provide the system with a look-up list of commodity terms. While there is substantial continuity over time in the materials that are globally traded as commodities, it is difficult to work with a modern list of commodity terms as they include many things that did not exist, or were not widely traded, in the nineteenth century. There are also a relatively large number of commodities traded in the nineteenth century that are no longer used, including a range of materials for dyes and some nineteenth century drugs. As a result, we set out to develop a new lexicon of commodities traded in the nineteenth century.

Before discussing in detail the methods that we used, it is useful to consider some of our requirements. First

[6] http://parlipapers.chadwyck.co.uk/home.do

[7] http://www.proquest.co.uk

[8] http://eco.canadiana.ca

[9] http://plants.jstor.org/

[10] http://www.amdigital.co.uk

[11] http://www.jstor.org/

we wanted to be able to capture the fact that there can be multiple names for the same commodity; for example, rubber might be referred to in several ways, including not just *rubber* but also *India rubber*, *caoutchouc* and *caouchouc*. Second, we wanted to include a limited amount of hierarchical structure in order to support querying, both in the database interface and also in the visualisation process. For example, it ought be possible to group together *limes*, *apples* and *oranges* within a common category (or hypernym) such as `Fruit`. Third, we wanted the freedom to add arbitrary attributes to terms, such as noting that both nuts and whales are a source of oil.

These considerations argued in favour of a framework that had more structure than a simple list of terms, but was more like a thesaurus than a dictionary or linguistically-organised lexicon.[12] This made SKOS (Simple Knowledge Organization System—Miles and Bechhofer (2009)) an obvious choice for organising the thesaurus. SKOS assumes that the 'hierarchical backbone' of the thesaurus is organised around *concepts*. These are semantic rather than linguistic entities, and serve as the hooks to which lexical labels are attached. SKOS employs the Resource Description Framework (RDF)[13] as a representation language; in particular, SKOS concepts are identified by URIs. Every concept has a unique 'preferred' (or canonical) lexical label (expressed by the property `skos:prefLabel`), plus any number of alternative lexical labels (expressed by the property `skos:altLabel`). Both of these RDF properties take string literals (with an optional language tag) as values.

The graph in Figure 4 illustrates how SKOS allows preferred and alternative lexical labels to be attached to a concept such as `dbp:Natural_Rubber`. Figure 4 illustrates a standard shortening for URIs, where a prefix such as `dbp:` is an alias for the namespace `http://dbpedia.org/resource/`. Consequently `dbp:Natural\_Rubber` is an abbreviation that expands to the full URI `http://dbpedia.org/resource/Natural\_Rubber`. In an analogous way, `skos:` and `rdf:` are prefixes that represent namespaces for the SKOS and RDF vocabularies respectively.

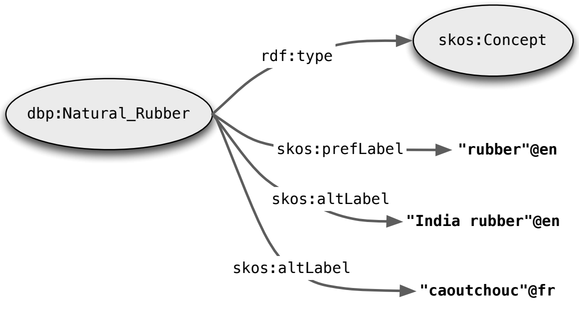


Figure 4: Preferred and alternative lexical labels in SKOS.

While a SKOS thesaurus provides a rich organisational structure for representing knowledge about our domain, it is not in itself directly usable by our text mining tools; a further step is required to place the `prefLabel` and `altLabel` values from the thesaurus into the XML-based lexicon structure required by the LT-XML2 toolkit during named entity recognition. We will discuss this in more detail in Section 3.2.

In the remainder of this section, we first describe how we created a seed set of commodity terms manually and then explain how we used it to bootstrap a much larger commodity lexicon.

3.1 Manual Curation from Archival Sources

We took as our starting point the records of the *Boards of Customs, Excise, and Customs and Excise, and HM Revenue and Customs* held at the National Archives.[14] They include a collation of annual ledger books listing all of the major goods, ranging from live animals to works of art, imported into Great Britain during any given year during the nineteenth century. These contain a wealth of material, including a list of the quantity and value of the commodities broken down by country. For the purpose of developing a list of commodities, we focused on the headings at the top of each page, drawing on the four books of the 1866 ledgers, which were the most detailed year available.[15] All together, the 1866 ledgers listed 760 different import categories. This data was manually transferred to a spreadsheet in a manner which closely reflected the original, and a portion is illustrated in Figure 5. In *Trading Consequences* we restricted our analysis to raw materials or lightly processed commodities and thereby discarded all commodities which did not fit this definition.

The two major steps in converting the Customs Ledger records into a SKOS format were (i) selecting a string to serve as the SKOS `prefLabel`, and (ii) associating the `prefLabel` with an appropriate semantic concept. Both these steps were carried out manually.[16]

For obvious reasons, we wanted as far as possible to use an existing ontology as a source of concepts. We initially experimented with UMBEL,[17] an extensive upper ontology in SKOS format based on OpenCyc (Matuszek et al., 2006). However UMBEL's coverage of relevant plants and botanical substances was poor, lacking

[12] The Lemon lexicon model (McCrae et al., 2010) is based on SKOS, but its richer structure, while linguistically well motivated, is more complex than we require for our application.

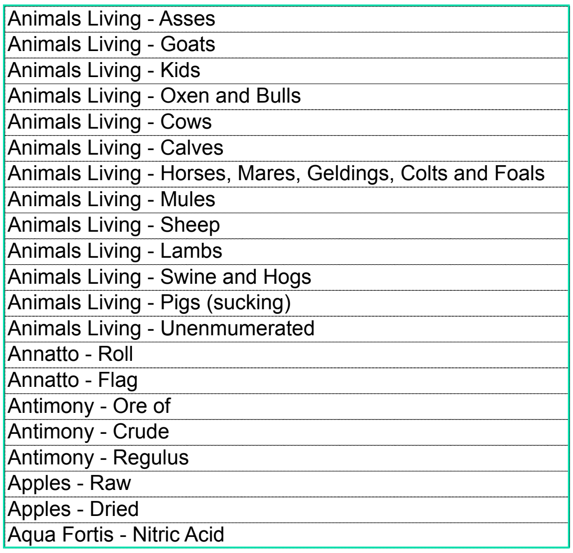
[13] `http://www.w3.org/RDF/`

[14] `http://discovery.nationalarchives.gov.uk/SearchUI/details?Uri=C67`

[15] The customs ledgers used for creation of the seed set of commodities is stored at The National Archives (collection CUST 5).

[16] Assem et al. (2006) present a methodology for converting thesauri to SKOS format, but the resources that their case studies take as a starting point are considerably more extensive and richly structured than the data we discuss here.

[17] `http://umbel.org`



Animals Living - Asses
Animals Living - Goats
Animals Living - Kids
Animals Living - Oxen and Bulls
Animals Living - Cows
Animals Living - Calves
Animals Living - Horses, Mares, Geldings, Colts and Foals
Animals Living - Mules
Animals Living - Sheep
Animals Living - Lambs
Animals Living - Swine and Hogs
Animals Living - Pigs (sucking)
Animals Living - Unenmumerated
Annatto - Roll
Annatto - Flag
Antimony - Ore of
Antimony - Crude
Antimony - Regulus
Apples - Raw
Apples - Dried
Aqua Fortis - Nitric Acid

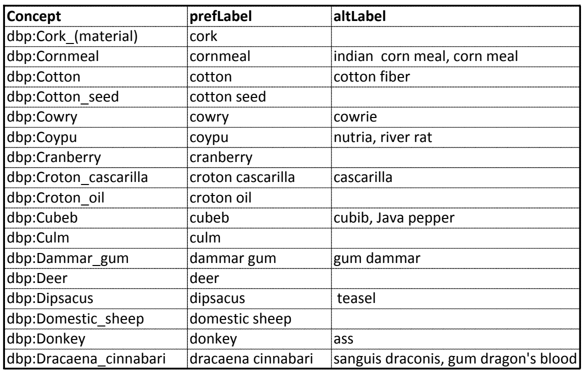
Figure 5: Sample spreadsheet entries derived from 1866 Customs Ledger.

for instance entries for *alizarin*, *bergamot* and *Dammar gum*, amongst many others. We eventually decided instead to base the ontology component of the lexicon on DBpedia (Bizer et al., 2009; Mendes et al., 2012), a structured knowledge base whose core concepts correspond to Wikipedia pages, augmented by Wikipedia categories, page links and infobox fields, all of which are extracted as RDF triples.

Figure 6 illustrates a portion of the converted spreadsheet, with columns corresponding to the DBpedia concept (using `dbp:` as the URI prefix), the `prefLabel`, and a list of `altLabels`. Note that *asses* has been normalised to a singular form and that it occurs as an `altLabel` for the concept `dbp:Donkey`. This data (in the form of a CSV file)[18] provides enough information to build a rudimentary SKOS thesaurus whose root concept is `tc:Commodity`.[19] The following listing illustrates a portion of the thesaurus for *donkey*.[20]

```
dbp:Donkey
   a        skos:Concept ;
   skos:prefLabel "donkey"@en ;
   skos:altLabel "ass"@en ;
   skos:broader tc:Commodity ;
   prov:hadPrimarySource
          "customs records 1866" .
```

Translated into plain English, this says: `dbp:Donkey` is a `skos:Concept`, its preferred label is `"donkey"`, its alternative label is `"ass"`, it has a broader concept `tc:Commodity`, and the primary source of this information (i.e., its provenance) are the customs records of 1866. Once we have an RDF model of the thesaurus, it becomes straightforward to carry out most subsequent processing via query, construct and update operations in SPARQL (Prud'Hommeaux and Seaborne, 2008; Seaborne and Harris, 2013), the standard language for querying RDF data.



Concept	prefLabel	altLabel
dbp:Cork_(material)	cork	
dbp:Cornmeal	cornmeal	indian corn meal, corn meal
dbp:Cotton	cotton	cotton fiber
dbp:Cotton_seed	cotton seed	
dbp:Cowry	cowry	cowrie
dbp:Coypu	coypu	nutria, river rat
dbp:Cranberry	cranberry	
dbp:Croton_cascarilla	croton cascarilla	cascarilla
dbp:Croton_oil	croton oil	
dbp:Cubeb	cubeb	cubib, Java pepper
dbp:Culm	culm	
dbp:Dammar_gum	dammar gum	gum dammar
dbp:Deer	deer	
dbp:Dipsacus	dipsacus	teasel
dbp:Domestic_sheep	domestic sheep	
dbp:Donkey	donkey	ass
dbp:Dracaena_cinnabari	dracaena cinnabari	sanguis draconis, gum dragon's blood

Figure 6: Customs Ledger data converted to SKOS data types.

3.2 Bootstrapping the Lexicon

The process just described allows us to construct a small 'base' SKOS thesaurus containing 319 concepts. However it is obviously a very incomplete list of commodities, and by itself would give us poor recall in identifying commodity mentions. Many kinds of product in the Customs Ledgers included open ended subcategories (i.e., *Oil - Seed Unenumerated* or *Fruit - Unenumerated Dried*). Similarly, while the ledgers provided a comprehensive list of various gums, they only specified *anchovies*, *cod*, *eels*, *herrings*, *salmon* and *turtle* as types of fish, grouping all other species under the 'unenumerated' subcategory.

One approach to augmenting the thesaurus would be to integrate it with a more general purpose SKOS upper ontology. In principle, this should be feasible, since merging two RDF graphs is a standard operation. However, trying this approach with UMBEL threw up several practical problems. First, UMBEL includes features that go beyond the standard framework of SKOS and which made graph merging harder to control. Second, this technique made it extremely difficult to avoid adding a large amount of information that was irrelevant to the domain of nineteenth century commodities.

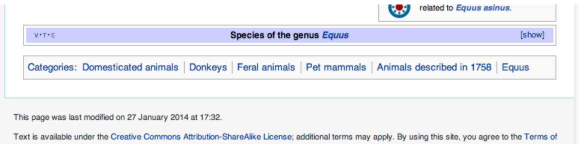
Our second approach also involved graph merging, but tried to minimise manual intervention in determining which subparts of the general ontology to merge into. We have already mentioned that one of our original motivations for adopting SKOS was the presence of a concept hierarchy; nevertheless, we had little need for a multi-layered hierarchy of the kind found in many

[18] Together with other resources from *Trading Consequences*, the word list is available as `base_lexicon.csv` from the Github repository `https://github.com/digtrade/digtrade`.

[19] The conversion from CSV to RDF was carried out with the help of the Python `rdflib` library (`https://rdflib.readthedocs.org`).

[20] The prefixes `tc:` and `prov:` are aliases for `http://vocab.inf.ed.ac.uk/tc/` and `http://www.w3.org/ns/prov\#` respectively.

upper ontologies. In addition to a class hierarchy of the usual kind, DBpedia contains a level of *category*, derived from the categories that are used to tag Wikipedia pages. Figure 7 illustrates categories, such as *Domesticated animals*, that occur on the page for *donkey*. We believe that such Wikipedia categories provide a useful and (for our purposes) sufficient level of abstraction for grouping together the 'leaf' concepts that correspond to lexical items in the SKOS thesaurus (e.g., a concept like `dbp:Donkey`). Within DBpedia, these categories are contained in the namespace `http://dbpedia.org/resource/Category:` (for which we use the alias `dbc:`) and are related to concepts via the property `dcterms:subject`. Given that the concepts in our base SKOS thesaurus are drawn from DBpedia, it is simple to augment the initial SKOS thesaurus G in the following way: for each leaf concept L in G, augment G with a new triple of the form $\langle L$ `skos:broader` $C\rangle$ (i.e., L has broader concept C) whenever L belongs to category C in DBpedia. To illustrate, given our `Donkey` example above, we would supplement it with the following triple:



related to *Equus asinus*.

V·T·E Species of the genus *Equus* [show]

Categories: Domesticated animals | Donkeys | Feral animals | Pet mammals | Animals described in 1758 | Equus

This page was last modified on 27 January 2014 at 17:32.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of

Figure 7: Wikipedia categories at the bottom of the page for `Donkey`.

```
dbp:Donkey
    skos:broader dbc:Domesticated_animal
```

We can retrieve all of the categories associated with each leaf concept by sending a federated query that accesses both the DBpedia SPARQL endpoint and a local instance of the Jena Fuseki[21] server which hosts our SKOS thesaurus. Since some of the categories recovered in this way were clearly too broad or out of scope, we manually filtered the list down to a set of 355 categories before merging the new triples into the base thesaurus.

Our next step also involved querying DBpedia, this time to retrieve all new concepts C which belonged to the categories recovered in the first step; we call this *sibling acquisition*, since it allows us to find siblings of leaf concepts that are children of the Wikipedia categories already present in the thesaurus. The key steps in the procedure are illustrated in Figure 8 (where the top node is the root concept in the SKOS thesaurus, viz. `tc:Commodity`). To continue our earlier example, the presence of `dbc:Domesticated_animal` in the hierarchy triggers the addition of concepts for animals such as camel, llama and water buffalo. Given a base thesaurus with 319 concepts, sibling acquisition expands the thesaurus to a size of 17,387 concepts.[22] This query-based methodology contrasts with, though is potentially complementary to, a machine learning approach to bootstrapping named entity systems as described, for example, by Kozareva (2006).

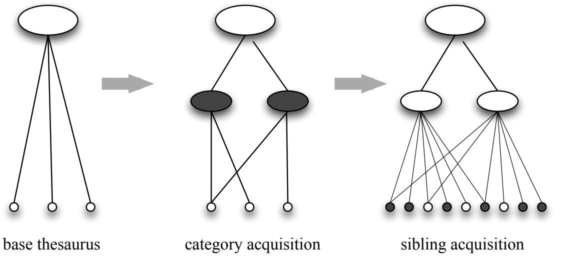


Figure 8: Sibling acquisition. A base thesaurus is augmented with new categories (indicated as black ovals), and these in turn lead to the addition of new leaf concepts (indicated as black circles) which they are broader than.

We mentioned earlier that in order for LT-TTT2 to identify commodity mentions in text, it is necessary to convert our SKOS thesaurus into an XML-based lexicon structure. A fragment of such a lexicon is illustrated in Figure 9. The preferred and alternative lexical labels are represented via separate entries in the lexicon, with their value contained in the `word` attribute for each entry. The concept and category information is stored in corresponding attribute values; the pipe symbol (|) is used to separate multiple categories. We have already seen that alternative lexical labels will include synonyms and spelling variants (e.g., *chinchona* versus *cinchona*). The set of alternative labels associated with each concept was further augmented by a series of postprocessing steps such as pluralisation; hyphenation and dehyphenation (*cocoa nuts* versus *cocoa-nuts* versus *cocoanuts*; and the addition of selected head nouns to form compounds (*apple* > *apple tree*, *groundnut* > *groundnut oil*). Such variants are also stored in the lexicon as separate entries. The resulting lexicon contained 20,476 commodity terms.

During the recognition step, we perform case-insensitive matching against the lexicon in combination with context-dependent rules to decide whether or not a given string is a commodity; the longest match is preferred during lookup. Linguistic pre-processing is important in this step — for example, we exclude word tokens tagged as verb, preposition, particle or adverb in the part-of-speech tagging. As each lexicon entry is associated with a DBpedia concept and at least one category, both types of information are added to the extracted entity mentions for each successful match, thereby linking the text-mined commodities to the hierarchy present in the *Trading Consequences* commodity thesaurus.

[21] `http://jena.apache.org/documentation/serving_data/`

[22] We accessed DBpedia via the SPARQL endpoint on 16 Dec 2013, which corresponds to DBpedia version 3.9.

```
<lex>
  ...
  <lex category="Rubber|Nonwoven_fabrics" concept="Natural_rubber" word="caoutchouc"/>
  <lex category="Rubber|Nonwoven_fabrics" concept="Natural_rubber" word="indian rubber"/>
  <lex category="Rubber|Nonwoven_fabrics" concept="Natural_rubber" word="rubber"/>
  ...
</lex>
```

Figure 9: Lexicon entries for the example presented in Figure 4.

4 Evaluation

4.1 Methodology

The quality of text mining software is often evaluated intrinsically in terms of the precision, recall and balanced F-score of its output compared to a human annotated gold standard. We also use this methodology to gain a better understanding of the quality of the commodity lexicon. We therefore prepared a gold standard by randomly selecting 25 documents extracts from each of the five collections listed in Table 1. Since many of the documents were too long to annotate in their entirety, we split each file into sub-sections of equal size (5000 bytes) and randomly selected one subsection per document containing one or more commodities and commodity-location relations. This resulted in a set of 125 files which we divided into a pilot set of 25 documents (5 per collection) and a main annotation set of 100 documents (20 per collection).

Annotator 1 was provided with guidelines on marking up entities and relations, and was asked to annotate the 25 pilot documents using the BRAT annotation tool (Stenetorp et al., 2012).[23] After an opportunity to clarify any issues, Annotator 1 carried out the main annotation by correcting the system output and adding any information that was missed by the text mining component. We refer to the resulting human-annotated dataset as the *gold standard* and compare our system output against it. Table 2 shows that relative to our gold standard annotations, the text mining prototype, which uses the expanded commodity lexicon described in Section 3.2), identified commodity mentions with a precision (P) of 0.59, a recall (R) of 0.56 and an F-score of 0.57.

These scores are determined with a strict evaluation where each commodity mention identified by the system has to match the manually annotated mention exactly in terms of its boundaries and type to count as a true positive. As soon as one boundary differs — for example, if the annotator identified *palm* and the system identified *palm trees* —- the mis-match counts as both a false positive and a false negative. In order to understand how often the commodity extraction results in a boundary error, we also applied a lax evaluation where a true positive is counted if both boundaries match exactly; or if the left boundary differs and the right matches; or if the left boundary matches and the right differs. The improved scores for the lax evaluation listed in Table 2 show that boundary errors significantly impact on system performance, with an equally negative effect on recall and precision.

Table 2 also gives inter-annotator agreement (IAA) scores for 25% of the gold standard. IAA was calculated by comparing the markup of Annotator 1 with a second annotator (Annotator 2) for the same data. The strict and lax scores show that IAA is not particularly high (F=0.72 and F=0.80) for a task that we expected to be fairly easy and that boundary errors are also one of the reasons for the disagreement, albeit not to such a large extent as in the system evaluation. After having carried out some error analysis of the double-annotation, we realised that Annotator 2 had not completely understood our definition of commodity and had mistakenly included machinery and tools (e.g., *scissors*) as well as general terms related to commodities (e.g., *produce*). Annotator 2 also missed several relevant commodity mentions which Annotator 1 had correctly identified. For these reasons, Annotator 2's markup was ignored when evaluating the text mining output.

4.2 Analysis and Lexicon Modification

When examining the output of the text mining prototype, we found that it had identified a total of 31,169,104 commodity mentions (tokens) across all five collections. However, these corresponded to only 5,841 different commodity terms (types). Since the *Trading Consequences* thesaurus contains 20,476 commodity terms, only 28.5% of the content in the lexicon corresponds to identifiable commodity mentions in the text. The top 1,757 most frequent commodity terms occur at least 100 times in our data; they make up a total of 31,113,978 commodity mentions in the text and therefore amount to 99.8% of all commodity mentions found. Figure 10 presents the average frequency distribution of different commodity terms (separated into bins) across all text collections.

The difference between the strict and lax boundary evaluations described above provide evidence that some of the commodity mentions in text were substrings of commodity terms in the lexicon (e.g., *seal* vs. *sealskins*) and vice versa. A detailed error analysis showed that incorrect and missing entries in the lexicon further decrease precision and recall, respectively, and OCR errors occurring in the commodity terms in the

[23] The pilot data is not included in the gold standard that is used for the evaluation.

	Evaluation	TP	FP	FN	P	R	F-score
Text Mining Prototype	**Strict**	**616**	**431**	**491**	**0.59**	**0.56**	**0.57**
	Lax boundaries	791	256	316	0.76	0.71	0.73
IAA	**Strict**	**283**	**112**	**109**	**0.72**	**0.72**	**0.72**
	Lax boundaries	314	81	80	0.78	0.80	0.80

Table 2: Precision (P), recall (R) and F-score figures for evaluating the performance of the commodity recognition prototype, as well as numbers of true positive (TP), false positive (FP) and false negative (FN) mentions. These figures are compared against equivalent inter-annotator agreement (IAA) scores in 25% of the gold standard documents. We provide evaluation scores for strict and lax boundary matching of entity mentions.

	Evaluation	TP	FP	FN	P	R	F-score
Text Mining Prototype	**Strict**	**616**	**431**	**491**	**0.59**	**0.56**	**0.57**
	Lax	791	256	316	0.76	0.71	0.73
(i) Removal of lexicon errors	Strict	603	331	504	0.65	0.54	0.59
	Lax	765	169	342	0.82	0.69	0.75
(ii) Context Rules	Strict	664	483	443	0.58	0.60	0.59
	Lax	777	370	330	0.68	0.70	0.69
(iii) Bigram-based additions	Strict	673	441	434	0.60	0.61	0.61
	Lax	855	259	252	0.77	0.77	0.77
Modified Lexicon: combination of (i)–(iii)	**Strict**	**652**	**353**	**455**	**0.65**	**0.59**	**0.62**
	Lax	792	213	315	0.79	0.72	0.75

Table 3: Precision (P), recall (R) and F-score figures for evaluating the performance of the commodity recognition prototype compared to the same scores for two optimisation steps. We provide evaluation scores for strict and lax boundary matching of entity mentions.

text also considerably reduce recall (Alex and Burns, to appear). In our gold standard, 9.1% (101 of 1,107) of all manually annotated commodity mentions contain one or more OCR errors. In order to improve the accuracy of the lexicon, we carried out three modifications, which are described below.

Step (i): Removal of errors from lexicon All commodity terms below that of rank 1,757 (in bin 1,701–1,800 and subsequent bins) have a frequency of less than 100. In *Trading Consequences* we are particularly interested in frequently occurring commodities as we aim to identify trends in trade. Consequently one of the authors of this paper (an environmental historian) manually checked the correctness of the top 1,757 commodity terms. 84 of them (4.8%) were considered to be errors (either real errors, OCR errors, commodities outside our scope, or overly-ambiguous terms) and were therefore deleted from the lexicon.

We then tested the effect this change had on the performance for against the gold standard. The scores in Table 3 show that step (i), deleting incorrect entries from the lexicon, has an expected positive effect on precision, which increased by 0.06 (to P=0.65). It also resulted in a small decrease in recall since Annotator 1 had marked several instances of the word *bread* as commodity mentions, which is arguably at the boundary of our definition of 'natural resources or lightly processed commodities'. He had also annotated *paper* and *linen* as commodity mentions, which are not within our definition. Eliminating incorrect terms from lexicon does not reduce the number of boundary errors made by the prototype, and consequently the lax boundary evaluation still results in an increase of 0.16 in F-score compared to the strict evaluation (F=0.59 versus F=0.75), the same as is the case for the prototype.

Step (ii): Context rules Having examined the boundary errors made by the prototype, we also applied rules to extend commodity mentions to the left or right in certain contexts. We shift a boundary to the left if a recognised commodity mention is preceded by a noun or proper noun starting with an uppercase letter or if it is preceded by another commodity mention. This boundary shift is carried out to capture noun phrases in which the recognised commodity mention is a head noun which is then specified further by its immediate left context (e.g., *coffee* is extended to *Liberica coffee* or *oil* is combined with *coconut* to yield *coconut oil*). We shift a boundary to the right in the case where a recognised commodity is followed by the word *tree* or *trees* (e.g., *palm trees*). We tested the effect of applying these context rules to the prototype (see step (ii) in Table 3). While this post-processing step decreases precision very slightly, recall increases by 0.4.

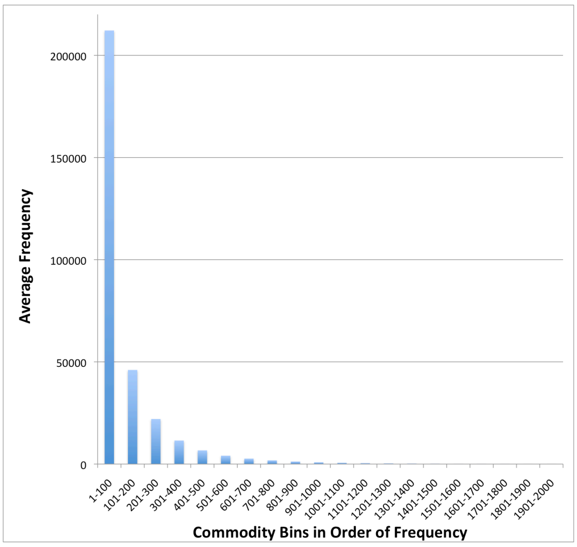
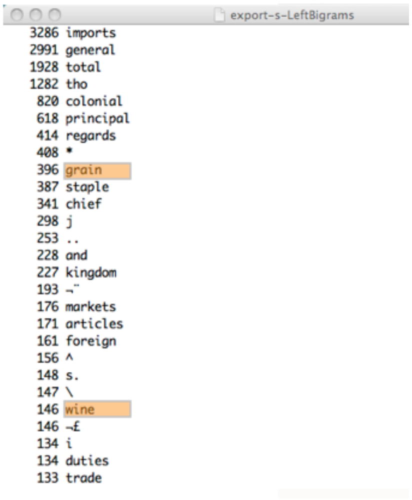


Figure 10: Average frequency distribution of different commodity terms split into bins of size 100. The *Trading Consequences* data contains a total of 5,841 different commodity terms. The graph is capped at the most frequent 2,000 terms as it would otherwise show a long invisible tail of very low average frequencies.

Step (iii): Bigram-based additions Finally, we conducted a frequency-based bigram analysis for a set of trade-related terms like *import*, *export*, *farm*, *plantation* of the text-mined collections (see an example in Figure 11). We manually examined frequently occurring left and right contexts of such words with the aim of identifying a list of terms for commodities of importance in the nineteenth century but which were not already contained in the lexicon and were therefore missed by the text mining. We identified a list of 294 commodity terms (including plural forms and spelling variants) which we added to the lexicon. Step (iii) in Table 3 shows that this change increases recall by 0.05 and precision by 0.01. When combining steps (i)–(iii), we obtain the highest overall F-score of 0.62 with the strict evaluation.

5 Conclusion

In many named entity recognition tasks, there is reasonable agreement in advance about the ontological scope of a given class. For example, when identifying mentions of people, locations, companies or dates in a corpus, we are not in doubt as to what constitutes these classes. By contrast, in the *Trading Consequences* project, our goal was precisely to gain a better understanding of what counted as a traded commodity during the nineteenth century. In other words, we were not only bootstrapping a lexicon, but were also trying to bootstrap the ontological class 'commodity' that was true for a specific time period. Given a small number of clear cases extracted from customs records, we used the categorial similarity of other entities to our seed set as means of extrapolating to a much larger set of candidate commodities. However, it is only when these candidates can be found as mentions in our corpus that we gain confidence in the belief that we really have identified new commodities. From the perspective of historical inquiry, progressing from around a dozen or so well-studied commodities in nineteenth century trade to around 2,000 is a significant step forward.



```
3286 imports
2991 general
1928 total
1282 tho
 820 colonial
 618 principal
 414 regards
 408 •
 396 grain
 387 staple
 341 chief
 298 j
 253 ..
 228 and
 227 kingdom
 193 ¬¨
 176 markets
 171 articles
 161 foreign
 156 ^
 148 s.
 147 \
 146 wine
 146 ¬£
 134 i
 134 duties
 133 trade
```

Figure 11: Most frequent tokens followed by the word *export* or *exports* found in the text-mined output of the HCPP data. This list excludes all occurrences where the left context is already recognised as a commodity. The commodities *grain* and *wine* have been marked by an expert historian as commodities that are missing from the lexicon.

The process of sibling acquisition via SPARQL query to DBpedia is a novel contribution, as far as we are aware, and we have argued that it can help to generate a lexicon which can be used as part of standard techniques in natural language processing. Although computational linguists are still relatively unfamiliar with RDF as a data model, we believe that its flexibility make it well suited to capturing the combination of lexical and encyclopaedic knowledge that is central to the digital history research described here. In addition, by basing our concepts on DBpedia, the 'linking' aspect of Linked Data (Heath and Bizer, 2011) gives us the potential to connect our commodity thesaurus to a wealth of other sources of knowledge about commodities.

Acknowledgments

The *Trading Consequences* project is a Digging Into Data II project (CIINN01) funded by Jisc, AHRC and SSHRC. It is a collaboration between the School of Informatics at the University of Edinburgh, EDINA, SACHI at the University of St. Andrews, together with York University and the University of Saskatchewan in Canada. We are grateful to Kate Byrne and to the three reviewers for their insightful comments on the paper.

References

Beatrice Alex and John Burns. to appear. Estimating and rating the quality of optically character recognised text. In *Proceedings of DATeCH 2014*.

Bea Alex, Claire Grover, Ewan Klein, and Richard Tobin. 2012. Digitised historical text: Does it have to be mediOCRe? In *Proceedings of the LThist 2012 workshop at KONVENS 2012*, pages 401–409.

Mark Assem, Véronique Malaisé, Alistair Miles, and Guus Schreiber. 2006. A method to convert thesauri to SKOS. In York Sure and John Domingue, editors, *The Semantic Web: Research and Applications*, volume 4011 of *Lecture Notes in Computer Science*, pages 95–109. Springer Berlin Heidelberg.

Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. 2009. Dbpedia — a crystallization point for the web of data. *Web Semantics*, 7(3):154–165, September.

William Cronon. 1991. *Natures Metropolis: Chicago and the Great West*. W. W. Norton, New York.

Gregory T Cushman. 2013. *Guano and the Opening of the Pacific World: A Global Ecological History*. Cambridge University Press, Cambridge.

Claire Grover, Sharon Givon, Richard Tobin, and Julian Ball. 2008. Named entity recognition for digitised historical texts. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 1343–1346, Marrakech, Morocco.

Tom Heath and Christian Bizer. 2011. *Linked Data: Evolving the Web into a Global Data Space*. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool.

Harold Innis and Daniel Drache. 1995. *Staples, Markets, and Cultural Change Selected Essay*. McGill-Queens University Press, Montreal.

Zornitsa Kozareva. 2006. Bootstrapping named entity recognition with automatically generated gazetteer lists. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, EACL '06, pages 15–21, Stroudsburg, PA, USA.

Daniel Lopresti. 2008. Optical character recognition errors and their effects on natural language processing. In *Proceedings of the second workshop on Analytics for Noisy Unstructured Text Data*, pages 9–16.

Cynthia Matuszek, John Cabral, Michael J Witbrock, and John DeOliveira. 2006. An introduction to the syntax and content of Cyc. In *AAAI Spring Symposium: Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, pages 44–49.

Stuart McCook. 2006. Global rust belt: *Hemileia Vastatrix* and the ecological integration of world coffee production since 1850. *Journal of Global History*, 1(2):177–195.

John McCrae, Guadalupe Aguado de Cea, Paul Buitelaar, Philipp Cimiano, Thierry Declerck, Asunción Gómez Pérez, Jorge Gracia, Laura Hollink, Elena Montiel-Ponsoda, Dennis Spohr, and Tobias Wunner, 2010. *The Lemon Cookbook*. The Monnet Project. `http://lemon-model.net/lemon-cookbook.pdf`.

P.N. Mendes, M. Jakob, and C. Bizer. 2012. DBpedia: A multilingual cross-domain knowledge base. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2012)*.

Alistair Miles and Sean Bechhofer. 2009. SKOS simple knowledge organization system reference. W3C recommendation, W3C, August. `http://www.w3.org/TR/2009/REC-skos-reference-20090818/`.

E. Prud'Hommeaux and A. Seaborne. 2008. Sparql query language for rdf. *W3C working draft*, 4(January).

Henry Nicholas Ridley. 1912. *Spices*. London, Macmillan and co. Ltd.

Andy Seaborne and Steven Harris. 2013. SPARQL 1.1 query language. W3C recommendation, W3C, March. `http://www.w3.org/TR/2013/REC-sparql11-query-20130321/`.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. BRAT: A web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning*, CONLL '03, pages 142–147, Stroudsburg, PA, USA.

John Tully. 2009. A victorian ecological disaster: Imperialism, the telegraph, and gutta-percha. *Journal of World History*, 20(4):559–579.