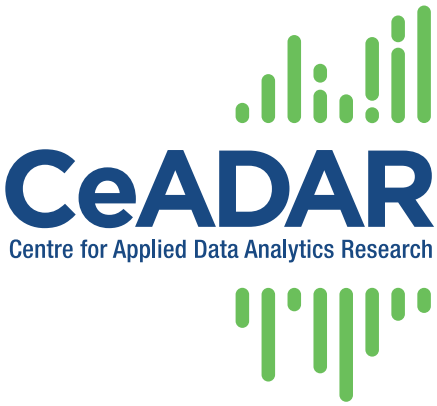

CeADAR – Centre for Applied Data Analytics Research

Enterprise Ireland Data Analytics Technology Centre

DocoPool – Harnessing Document Knowledge – Technical Specification

Document Type:	Technical Specification
Project Title:	CeADAR
DDN Theme:	1 – Intelligent Analytic Interfaces
Theme Leader:	Sarah Jane Delany (DIT)
DDN Sub-Theme:	Ease of Interaction
Authors:	Caroline Maillet, Susan McKeever
Document Version:	0.3
Date of Delivery to ISG:	September, 2015
Number of pages:	10

ABSTRACT

In this report we provide a Technical Specification for the DocoPool demonstrator system, part of the CeADAR Ease of Interaction theme. The purpose of the DocoPool system is to glean knowledge from pools of text-based documents. The core functionality of the DocoPool demonstrator is the automatic identification of topics within a document pool. Additional use cases (location extraction, phrase viewing and synonym searching) will be included subject to time availability.

CeADAR is a research partnerhip comprising University College Dublin, University College Cork, and Dublin Institute of Technology.

http://www.ceadar.ie

Contents

1 Description of Industry Needs

Many companies process and store discrete documents as part of their internal and customer facing processes. Word processed documents, such as those with .doc(x) and .pdf file extensions, are common document types. Unlike structured data held in a database, these document types contain data in an unstructured text format, held in isolation within the document. When a company needs to analyse groups of these documents, substantial manual effort may be required to analyse each document separately.

The purpose of the DocoPool system is to support the analysis of these pools of documents. To narrow down the requirements, we discussed a number of relevant use cases with our industry partners. Each use case is outlined with an example as follows:

- *The ability to automatically identify the topics in a pool of documents:* A business user may need to analyse the content of a set of documents in order to see the purpose of each document, grouped by topic. This could apply in the case of insurance claims documents, to determine if there is any particular unusual occurrence of theme words such as "asbestos", as part of a fraud detection process. Likewise, a news analyst could peruse documents to identify the topics covered, without having to read the content of every document.

- *Automatically identify locations:* This use case would support the automatic extraction of locations from a pool of documents, such as automatically extracting customer address locations or insurance claim event locations. Addresses, which are a type of "named entity", would be identified and extracted using *Named Entity Recognition* information extraction techniques.

- *The ability to view the occurrence of particular phrases:* This use case would enable a user to direct a search for particular phrases in a document pool, graphed over time. For example, a data analyst examining news stories could search for the occurrence of "property bubble", "recession", and "banking crisis" over a document pool, in order to see the trend over time of occurrence of these phrases. Such phrases, termed n-grams, will typically relate to particular trends or events of interest to a business.

- *Synonym searching:* Searching for particular words, with the automatic inclusion of the search word synonyms, would allow companies to do a comprehensive search on a pool of documents. For example, a data analyst searching for the word "ship" would also have search results that include synonyms of "ship". In this example, "ship" can be either a noun or a verb. The results returned to the user would need to allow for the alternative meanings of a word.

It is unlikely that all four uses cases described can be delivered within the development time allocated to the DocoPool demonstrator. We have prioritised, with guidance from industry, the order of delivery of use cases in order to deliver the more beneficial use cases first. Topic modelling is the first use case to be delivered. The remaining use cases have been prioritised

from highest to lowest in the following order: location extraction, n-gram viewing and synonym searching.

In the next section, we examine the systems that will be involved in the DocoPool demonstrator implementation.

2 Systems used

The use cases described use text mining techniques such as information retrieval, lexical analysis and topic modelling. These enable us to find patterns and meaning in natural language text. Various research institutions have published libraries and ontologies to enable text mining algorithms to be used by the wider research and business community. The DocoPool demonstrator will, where appropriate, make use of existing published software. The pre-existing libraries and systems to be used are outlined here. The detailed use of such systems, and their context within the DocoPool demonstrator are explained more fully in Section 3.

DocoPool will need to support the upload of individual documents from the document pool for processing. DocoPool will support the following two input formats (1) unstructured text files and (2) formatted word processing documents (.doc(x) and .pdf). All documents of type (2) will then be converted into text files. There are various candidate libraries to support document conversation such as the text mining (tm) package [7] in the statistical programming tool, R, and the python library, python-docx [1].

To support our text-mining tasks, we may use components of the Natural Language Toolkit (NLTK) [3], Stanford Natural Language Processing (NLP) [2] and scikit-learn machine learning package [6] to support the following:

- Pre-processing of uploaded text data, including *tokenisation*, *stemming*, and *stop word* removal. Tokenisation converts sentences and word streams into individual text tokens. Stemming reduces words to their root form, so that substitute word forms are identified. Stop words are those words such as "a", and "the" that do not enhance the ability to interpret meaning, so are not needed for some types of document text analysis.
- Topic modelling: for the topic modelling use case, processed text from each file or document will require transformation into a set of terms and their frequency of occurrence within each document, using a document term frequency matrix.
- Named Entity Recognition (NER): NER is an information extraction task that extracts and categorises specific entities (typically proper nouns such as names and locations) from text, using part-of-speech tagging.
- N-gram identification: n-grams, in the context of DocoPool, represent sets containing one or more words that occur in sequence within a document.
- Synonyms search: synonyms of users input keywords can be retrieved with the use of an ontology, such as WordNet [5].

Use case(s) delivered in the demonstrator will require visualisation of results. We will use the JavaScript library D3.js [4] to support the development of visualisations.

The DocoPool demonstrator will use a web interface, with access to the demonstrator available from a remote site. Transferring large numbers of company documents over the internet has potential performance implications. Therefore, we may aim to architect the demonstrator so that the document processing is performed on the client side, without the need to transfer documents across the internet.

3 Approach

As shown by Figure 1, our approach to the DocoPool demonstrator can be explained as a series of steps as follows:

- Upload
- Pre-processing
- Individual Use case analysis and visualisation.

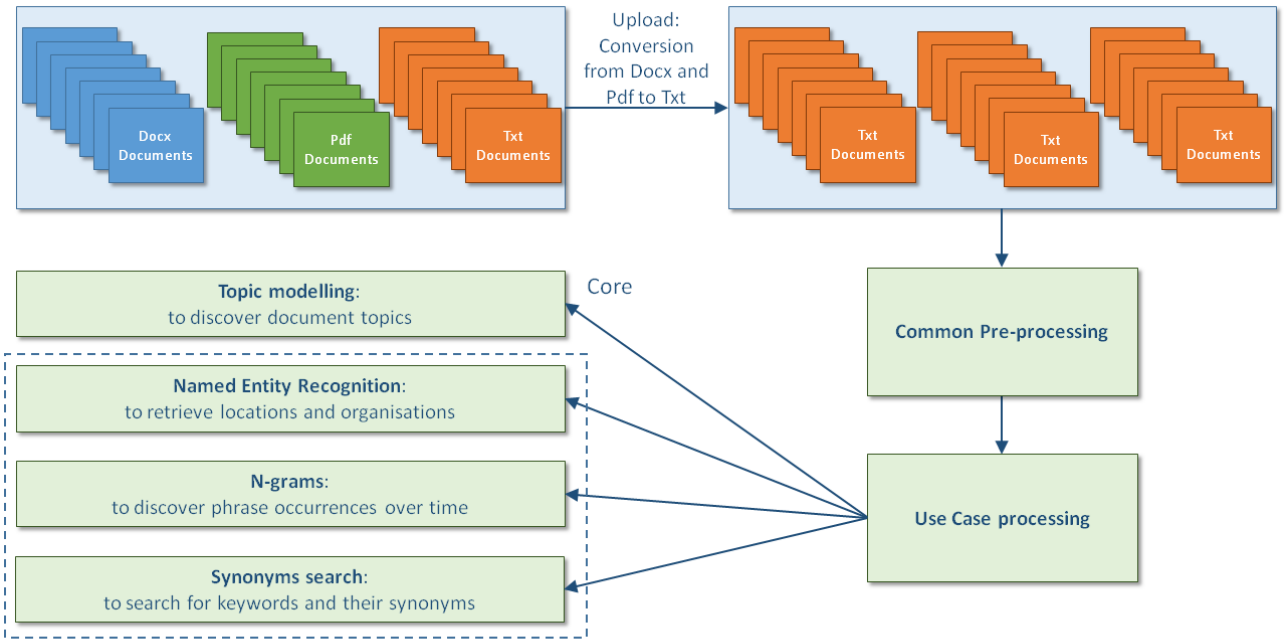

Figure 1: DocoPool's process flow.

3.1 Upload

The input to the upload step is the "document pool" i.e. the raw text files and word processing documents to be processed. Documents may have associated *properties* meta-data extracted from the file properties (such as author and date). Each document will have zero or one row of associated meta-data, where the link from document to meta-data is via the document name. We assume that each document file name is unique in the document pool. The input specification for file upload is explained in Section 5

Meta-data from company systems was discussed in detail with one of our industry partners. This type of custom meta-data can be uploaded to DocoPool for analysis, and should be uploaded as per the text file specification in Section 5.2. Each row of meta-data rows is

effectively a new document, so each row will be uploaded as an individual text file. These custom meta-data files will not be linkable to other documents, and will be included in the document pool for analysis as per standard text files.
DocoPool will allow the user to define a location for the document pool contents. Once the user triggers the upload, DocoPool will convert individual .doc(x) and.pdf files into individual text files. Post upload, all documents in the document pool will then be available as text files, in a common format. *Properties* meta-data will be held in a separate identifiable text file, traceable to the originating document via the document's file name.

3.2 Common pre-processing

When the upload of the document pool has completed, a set of pre-processing steps are required in order to convert the text files into a state suitable for analysis. Each use case has a specific set of pre-processing tasks. The common pre-processing step across all four use cases is the conversion of the text into separate tokens (tokenisation). The tokenised text files are then available as input to the specific use-case processing step.

3.3 Use-case processing and visualisation

The user will have one or more use cases to choose from, depending upon the number of use cases delivered in the demonstrator: (1) Analyse document topics (i.e. topic modelling) (2) Extract locations (3) View phrases (n-gram viewing) (4) Search for synonyms. The processing required for each use case is as follows:

3.3.1 Analyse document topics (topic modelling)

Stop words will be removed from the tokenised text files. Stemming will then be performed to remove redundant word forms. Each original document or text file will then exist as a "bag of words" representation within its own text file. Next, the text files will be converted into a single document term frequency matrix. Each document will be represented as a single row in the matrix, with word frequencies mapped against word columns. Properties meta-data will be also available as a separate, linked data structure, such that each row of meta-data is linkable to a particular row in the document term matrix.

Unsupervised (i.e. automated) topic modelling will be performed by analysing the frequency of words in a document, relative to how common that word is in the overall document pool. Topics are identified as groups of words that have occurred together in the same documents. Each topic is individually interpretable, providing a probability distribution over words to pick out a cluster of correlated terms. Topics extraction will be performed with the Non-negative Matrix Factorization (NMF) or Latent Dirichlet Allocation (LDA) algorithms.

To visualise the results, the user needs to be able to see the topics identified based on groups of words and the relative "size" of the topics across the document pool. Each topic will be clickable in order to drill down to the underlying documents that have been identified as contributing to the topic. A sample visualisation is show in Figure 2. The list of topics is shown as horizontal bars in a scrollable list. The relative popularity of the topic, in terms of

the number of documents in the topic, is indicated by the width of the horizontal bar.

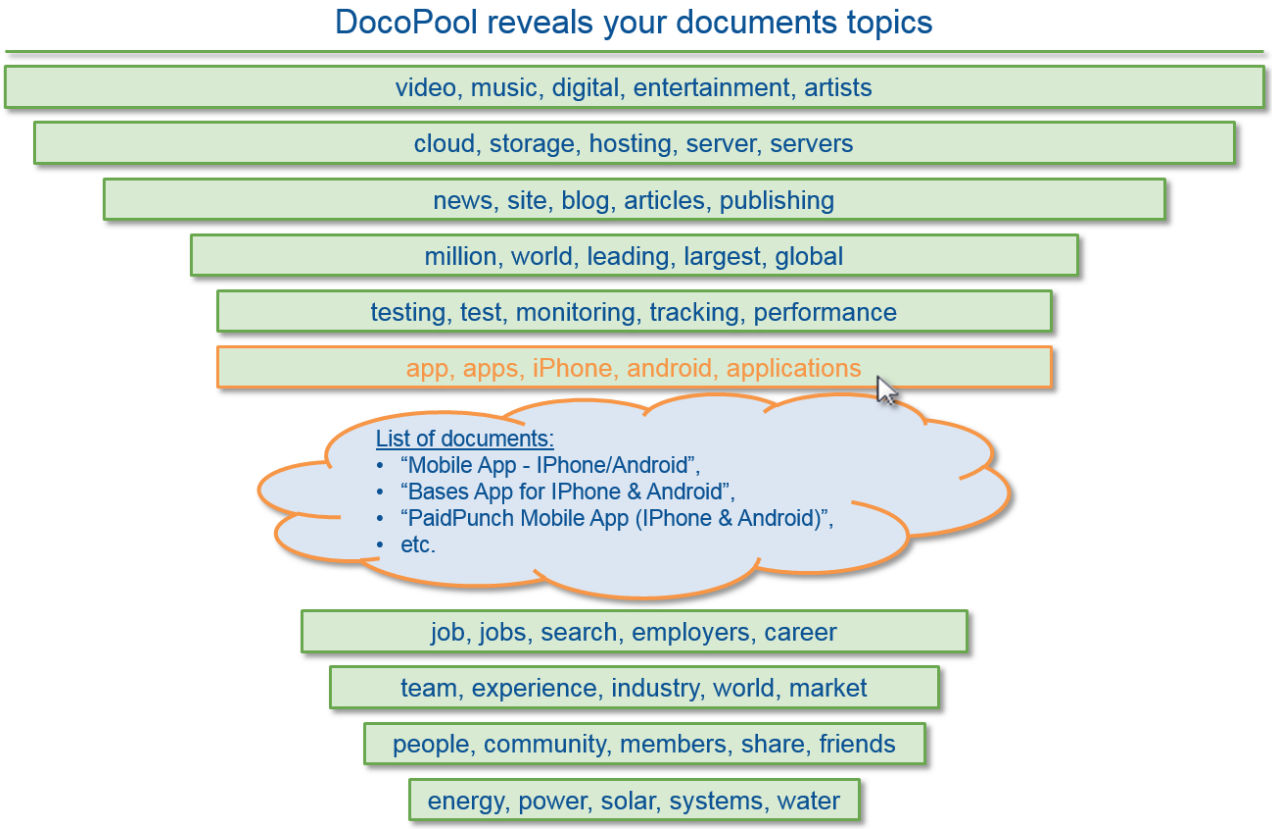

Figure 2: Sample visualisation for DocoPool's topic modelling.

3.3.2 Extract locations (NER)

For this use case, part-of-speech (POS) tagging will be performed on the tokenised text files so that each token can be identified according to its grammatical type (such as noun, verb, adjective). The files will then be parsed to extract the named entities, including locations.

The user will be presented back with a list of the locations, and the associated documents in which they occurred.

3.3.3 View phrases over time (n-gram viewing)

For this use case, the user will enter a word or phrase to search across the document pool. This use case assumes that the creation date of the document or text file is available as properties meta-data. The sequence of words in an n-gram is critical, so all adjacent tokens are potential n-grams. The system identifies the occurrence of the phrase in the tokenised text files and retrieves the relevant document creation dates from the properties meta-data.

The visualisation of the n-gram occurrences over time will be shown over a time frame, where the time frame ranges from the earliest document created date to the most recently created document in the document pool. The graph will be clickable to allow the user to drill into underlying documents.

3.3.4 Synonym search

For this use case, the user will specify a single search word to trigger a search across the document pool. Stop words will be removed from the tokenised files. POS tagging will then be performed on the remaining tokens to identify words that may have multiple meanings. Synonyms of the search term are retrieved from an ontology of synonyms.

The visualisation of the synonym search will consist of a list of documents that contain the search term and/or its synonyms, in a sortable list, where sorting can be done on document name or creation date.

4 Risks and Challenges

As stated in Section 1, the number of target use cases discussed in detail with industry partners is likely to be too large to deliver in a single demonstrator. Topic modelling will form the core functionality of the demonstrator. We will include location extraction, n-gram viewing and synonym search uses cases, in that order, according to the time available during demonstrator development. In addition, the following clarifications of scope should be noted:

- The upload will process all documents in the upload location. Incremental uploads of documents to extend an existing document pool that has already been processed are not supported.
- Scanned documents using Optical Character Recognition are not a supported input.
- Text embedded within images in the pool of documents is not included in the text analysis.

Analysing pools of documents presents many potential use cases. Discussions with our industry partners have identified other more company specific use cases that also involve extracting knowledge from text data. For example, some users would like to analyse public web content or to validate the correctness of content within documents, such as the completion of claim forms by users. The use cases collected are too varied and numerous to be supported by a single CeADAR demonstrator. They can be explored as potential future work with individual companies, post-demonstrator delivery.

The speed of performance of the demonstrator will depend upon several factors, including the number of documents in the pool, the performance capability of the client machine(s) if used, and the particular use case selected. Limitations on, or targets for performance are not specified in the demonstrator. We will monitor performance as we develop and test the demonstrator.

The accuracy of the demonstrator will be easier to evaluate and optimise if we can obtain real documents from companies to test. In the absence of sample documents, we will use our own document sets to test the demonstrator.

5 Appendix - Input file specifications

We define the input data format for the demonstrator for documents and text files. All documents and text files names must be unique in the document pool.

5.1 Document input

The following document types can be uploaded:

- .docx (Microsoft Word 2007 onwards),
- .pdf (Portable Document Format),

Words are assumed to be separated by one of the following cases:

- a blank space only,
- a punctuation mark from the list of word separators,
- a blank space and one or more punctuation marks in any sequence.

Punctuation marks that are treated as word separators:

- brackets: "[ ] ( ) { }",
- colon: ":",
- comma: ",",
- dash: "-",
- ellipsis: "...",
- exclamation mark: "!",
- full stop, period: ".",
- question mark: "?",
- semicolon: ";",
- inverted commas, quotation marks: " " ",
- slash, stroke: "/".

5.2 Text file input

DocoPool users will also be able to upload .txt (text files).
Words are assumed to have the same separation rules as those for document input in Section 5.1.
Company meta-data may be uploaded as a text file (.txt). Each meta-data field will be separated using the same word separators as text and document input data.

References

[1] Python-docx. `https://python-docx.readthedocs.org`.

[2] Stanford NLP. `http://nlp.stanford.edu/index.shtml`.

[3] Steven Bird. NLTK: the natural language toolkit. In *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July*, 2006.

[4] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3: Data-driven documents. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2011.

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

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[7] Stefan Theußl, Ingo Feinerer, and Kurt Hornik. A tm plug-in for distributed text mining in R. *Journal of Statistical Software*, 51(5):1–31, 2012.