Machine learning REDACTED Topic modelling and Document clustering for a client project library

Executive Summary

From SharePoint, we selected a large client where we have done several different engagements, leaving a sizeable document trail. We successfully machine-read all the different words in each document, and uncovered clusters of similar documents, document outliers, and the main topics covered. This should be applied and extended to reading project documents at a client and so understand their whole portfolio. That way we are able to combine a machine-reading view with our own assessment of the client's business.

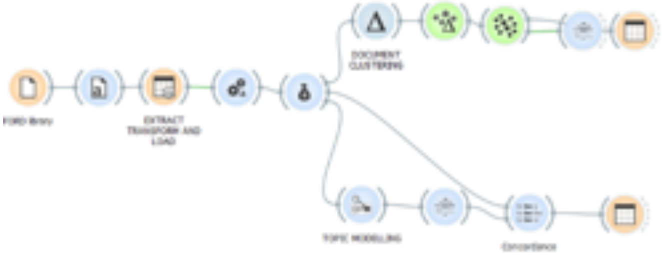
Introduction

We have taken a typical client library, in this case all xxx engagements. This exercise identifies client teams have actually been working on, to complement client interviews at the start of an engagement. This will help in deciding with the client which focus areas need most attention, by checking to see whether the client perceptions of the project challenge /opportunity are similar or different to the themes that are covered in their documents. We would like to know

i) the main clusters of documentation by content area

ii) any stand-out documents

iii) what topics are covered by the library.



Main results

i) the main clusters of documentation by content area

   a. Engineering and design technical documentation

   b. Management documentation around Sharepoint and the engagement

   c. Project lists and reports

ii) any stand-out documents: the Plan, a PM standards document and a CRUSH report.

iii) Dominant topics themes across your library are:

   a. xxx

   b. xxx

   c. xxx

Take away for discussion with the client

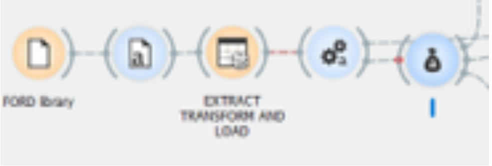
CLUSTERS: Do document clusters represent what you would expect to find in your documents? Are there any significant areas where you suspect there might not be enough work or projects?

THEMES: Are you comfortable that your projects altogether are appropriately reflecting the importance of the highlighted themes ? We could explore these themes with the relevant extracts from the respective Watson or Google Knowledge graphs, here:

Data

I have selected something which I know nothing about, so that I can see what insights I can get just from the analysis. On Sharepoint, I selected a client with multiple engagements over a period of years, with around 2000 documents in total. I have so far added extracted about 500 documents to study. It would be possible to add in the remaining PowerPoint files, emails and spreadsheets, but a little more pre-processing would be required. This library could be a project library or a process and standards library, or a collection of day to day management reports.

Extract, Transform and Load the data



1. Data extracted: The documents were converted into a text format compatible with Orange, and saved as one single libary. These documents are then read in from the file as a document corpus
2. Data transformation and load
3. sample the data, as many of the early results in this paper have been run with smaller subsets of the 500 documents for speed.
4. Pre-process the data . The first widget turns all the text into tokens (normally words).
5. The next widget turns each document into a "bag of words", showing how many words are used in each.

A lot of work can be done here to improve the way that the documents are tidied up and the words are selected.

Document clustering using distances between documents, then representing as a network

We first compute the distances between each document and every other document, based upon which words are found in which. There are several ways of visualising this data, and I have selected to show the distances as a network. This method allows you to treat documents that are like each other as being nodes connected with an edge, and we can set the distance threshold at which we decide to allocate an edge. When doing this, we are paying more attention to the clusters rather than to the outliers. By starting with clusters that are at a manageable human scale, such as three to six clusters, we can get a sense of the library. Then we can progressively allow the more ambiguous documents to appear and see more complex clusters only when we need it. Or we can have focussed in on one area with a manageable number of key documents, and only then start to find our way round the detailed clusters and relationships. This method could also be run on subsets of the data once we have decided which topics or words we are interested in. I.e. We could have selected a topic of interest and then looked in detail at the relevant documents as a network.

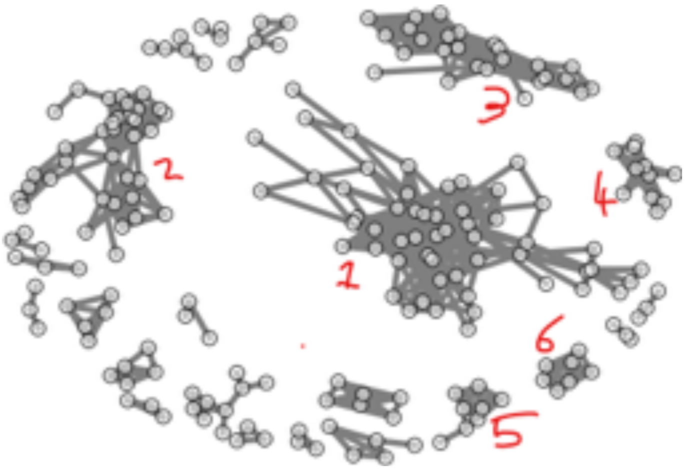
There appear to be large document clusters:

1. xx
2. xx

Looking at some of the smaller clusters, there is:

3. xx
4. xx

And so on. Once you select a cluster, you can view the documents and respective cloud.



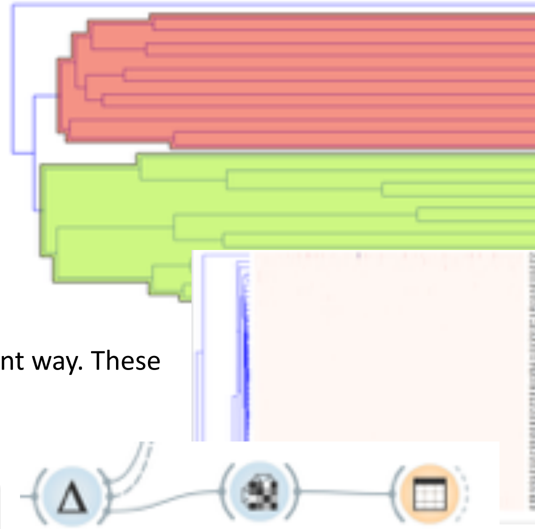
Looking for outliers, and testing our original clusters at the same time

We can do this with the whole library at once, but here I am showing a subset of the documents for speed.

Using the hashing algorithm, we establish different clusters of documents, based upon word-similarity. Here we see that the **xx**

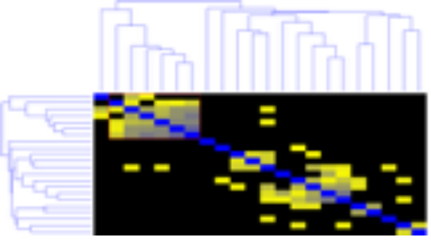
nd

stands out as being very different. There is also a cluster of English documents, as well as a cluster of German and SharePoint documents. This algorithm will scale well up to much large collections, and it also will be useful for the future, when trying to find a cluster of documents that is most relevant to a search document or search paragraph that we have. Once we select a cluster, however big or small, we can then view those documents together, or export them into their own folder.



The first time, it is worth looking at alternate clustering algorithms, to see which perform better, or whether they pick up different aspects of the library. This is a K-means algorithm, that separates out **a xx report, and a Project xxx.** as well as the previously identified plan

We can also use the distances as calculated earlier but visualise them in a different way. These here are a subset of the documents. We can see different clusters, both from the blue dendrogram, and from the yellow shading.



We can select each cluster and inspect. The top left cluster is a set of German SharePoint documents.

The small tight cluster in the middle is a set of Internet content documents.As above, where we show big red and green clusters, we could also look at the hierarchies of clusters like this, and going down as deep into the clustering as we wish. Here, a multi-dimensional scaling tries to reduce difference to two dimensions, and we see similar outliers.

GCG_010001_01EN_1105_PM-Standard_2012-08-21
Topic Modelling
Word Cloud

Topic modelling

Topic modelling is another alternative, where the advantage is that topics can overlap each other within one document, whereas the above document clustering doesn't allow this.

We can investigate one topic more closely. By selecting Topic 10, we see some detailed areas explored within this topic, and particular documents and sentences could be explored in the same way as for the main document word cloud above, drilling down into the concordance of sentences for each subject of interest.

Taking the results, and ignoring the first topic, which relates to numerical German lists, the top xx topics are:

1.   xx , 2. xxx

This is particularly powerful when we start to use broader knowledge graphs available to us. To probe topic 9 and 10 for example we can compare and combine with the Watson news knowledge graph. From the link page, we can read off each of the nodes on this graph, as well as the key articles. For topic 10, I have sent Google search for the client and China to output as a graph in Infranodus. Again, we can read create and tailor this knowledge graph.

First explorations of themes with the client

Stepping back to the actual process with the client, there is an advantage in placing in some intermediate analysis before the above. This helps to calibrate and judge expectations, as well as identify themes and pain points, and to get the client starting to work with us to direct the analysis to areas of the library that will shed light on the biggest opportunities and problems.

For example, word cloud can be explored with the client early in the engagement. Words can be selected, and sent forward to be viewed in a concordance, to show how the word is being used within the sentence. The best documents can be selected and sent to be viewed and saved separately. The client is likely to latch onto certain themes worthy of investigation, either because they confirm opinions or because they surprise. There may be work that has been done by the team which the client was not aware of in these key areas. This is also something that can be returned to later, perhaps doing a separate more detailed NLP exercise on the key documents once they have been identified. There are many ways of tailoring the pre-processing of text, loading it with key terms, which allows this to become a focused exercise where required.

With Sentiment analysis, we can inspect documents in on exceptionally positive or negative sentiment, like this Automotive resume that is low on negative and high on positive sentiment.

How could this be improved ?

A general point here: NLP is one of the fastest growing applications. I would say I am about five years behind with all this, but even the 5-year-old stuff here is pretty good. This example has been a good place to start, and to try and identify core workflows that get reasonable results. From there on, it will be possible to scale up 10, 100, 1000 times, to large projects, departments and then a whole enterprise. This is one of the key reasons why ML is so important at the moment. Even with the current use cases and scale shown here, these are useful for intelligent consultants in their next project, even just doing the project day job. I have been using similar methods myself in day-to-day service development, knowledge acquisition and information retrieval with my own personal and learning data. Going forward, we can build new project services to sell, as well as improve the way we run our own engagements and operations. There are also significant knowledge and Enterprise data that can be rapidly exploited. Our strong competitors will already be doing this, whereas some of our peers will not even know that this can be done, and this gives a 1-2-year time frame for competitive advantage.

There is nothing special about Orange. What we are doing here, we can do on a whole number of tools and platforms already. I am trialling and learning several of the obvious next systems at the moment. Orange is proving useful at the moment in order to help build understanding in our peers, because of the visual manipulatable interface. Sometimes it will be the right tool for the job, but some of us will be using whichever open source or enterprise tool is easiest for the given job, as we grasp the principles. These example workflows can quickly be moved across to the best tool.

Improvement	***Benefit***	***Status***
Remove poor tokens by inspection	Optimisation	Basic trials
Optimise pre-processing	Optimisation	Basic trials
Look for appropriate bigrams	Optimisation	1 successful trial
Document classification by supervised learning is straightforward if the client has already done some good librarian ship in defining sub folders.	A location and subject can be suggested for new documents.	Several successful trials on different data with better file structure. For example I have sussessfully predicted authors from 24000 Kindle annotations.
Go from 500 documents to 2000 documents	Scaling up	Have identified how to convert, but do not have time yet to do it myself
Incorporate Excel, going up to perhaps 3000	Scaling up	Have an idea how to combine but not tested
Tokenizing sentences rather than words	Knowledge retrieval	1 successful trial
Tune tokens with the IBM Watson Knowledge Graph for ...	Knowledge retrieval	Know how to do it
Tune tokens (in lexicon) through use of our project portfolio catalogue	Knowledge retrieval	Know how to do it
Using TensorFlow with a Character RNN	Deep learning should improve the results and get on the first step towards contextual natural language generation. This will be appropriate for a new use case of creating first drafts for technical proposals, when applied to	Investigating use of Uber's Ludwig library to implement