Stakeholder report

Authors: Katharina Granberg, Sergiu Ropota and Johannes Hoseth

The present report is written to explain the project focus, methods and results of the M2 group assignment. All code that has laid the ground for this report can be found in the attached notebooks. Four notebooks are attached, with this recommended reading order: (1) The data Scraping notebook (which is voluntary reading), (2) the Data Cleaning notebook, (3) the NLP notebook and (4) Network Analysis Notebook.

Introduction to project focus

The aim of our group work is to investigate the emergence and evolution of the concept of 'agility', as found in the management and business literature. The concept has emerged from the field of production optimization, then spread towards software development and, in recent years, has caught on in the business literature. The project focus is thought to be interesting and relevant in terms of applying both network analysis and NLP methodology. This is so because the scientific community of data science, expressed through the articles being published, can be both rich in semantic meaning and in relationships between articles and topics. This kind of data has its own name, *bibliographical data*.

The main research questions our project tackles are:

- What are the main theoretical pillars that research on 'organizational agility' builds on.
- Where is the academic discourse centered around today?

We chose to retrieve a data set ourselves on the topic, through the database Web of Science. Here, and like other scientific databases, it is possible to adjust one's search queries, and export the results in convenient .txt files.
With a search for both articles concerning either 'agility' or 'agile' and restricted to academic articles, published in business and management journals we retrieved a set of 1200 data points.

Prior more encompassing searches yielded far more results and have been obtained by scraping Web of Science to save time from manually exporting batches of 500 articles (due to inherent export restrictions set by the website).

However, as we later found out, processing a such big dataset (100k data points), turned out to be impossible to process given Google Colab's limited computational abilities.

Initial Data cleaning

Before performing any NLP or network specific operations, we first sought out to have a clean dataset, containing only the features of interest, such as 'abstract' and 'field', and only including articles with data for all the important features.
We chose to keep features that carry semantically important information about the articles, i.e. textual data, along with variables that can be used as identification entities within a network, e.g. authors and DOI. The head of the full DataFrame is shown below:

	Author	Title	Journal	Sub-field	Keywords	Abstract	References	City of pub	Year	Field	DOI
0	Ettlie, JE	R&D and global manufacturing performance	MANAGEMENT SCIENCE	R&D; agility; process innovation; product inno...	LARGE MULTIPRODUCT FIRMS; D INTENSITY; DIVERSI...	R&D intensity and manufacturing performance we...	Allison P., 1990, SOCIOL METHODOL, V20, P93, D...	LINTHICUM HTS	1998.0	Management; Operations Research & Management S...	10.1287/mnsc.44.1.1
1	Ottaway, TA; Burns, JR	Adaptive, agile approaches to organizational a...	DECISION SCIENCES	NaN	FLEXIBLE MANUFACTURING SYSTEMS	Intelligent agent-based approaches to software...	Anthony R.N., 1965, PLANNING CONTROL SYS; BUZA...	ATLANTA	1997.0	Management	10.1111/j.1540-5915.1997.tb01320.x

Natural Language Processing

We have chosen to focus the natural language processing part of the project on the abstracts, since they have a lot of information in the textual data. But being longer texts, we needed to do some NLP preprocessing.

We have,

- Converted all text into lowercase and removed numbers.
- Removed *stop words*, meaning word classes thought to be semantically irrelevant, such as determiners ('an', 'the') and adverbs ('however', 'both').
- Removed additional words based on their Part of Speech, like auxiliaries ('has', 'should'), should some of them not have been picked up by the stop words. We also use POS to remove punctuation.
- Converted all tenses of the words into their root words, which is called *lemmetization*. An example could be 'runs/running/ran' are being expressed as 'run'.
- And returned all the words as a list of tokens, so they are more easily analysed and used in machine learning.

Then we made a Bag-of-Words which is a good way of representing which words appear in each abstract, and how many times. Then we went a step further and used Term Frequency–Inverse Document Frequency, to scale the Bag-of-Words, to show how relevant for the abstracts the words are, that appear in the abstracts.
Then we used Latent Semantic Indexing to do dimensionality reduction and find topics. From this we made a document-topic matrix. Which we used later for some unsupervised learning.

We also made a word-embedding capable of, given a word used in the abstracts, finding similar words. We chose the Word2Vec model, which is fast and a bit more appropriate towards embedding academic language, compared to the otherwise more capable FastText model.

Then we did some simple EDA (Exploratory Data Analysis) which is yet another way of investigating our data through text classification. Here we found the most used words used in three different fields, highlighting how the fields differ. Right out the box, EDA proves to find quite meaningful fields.

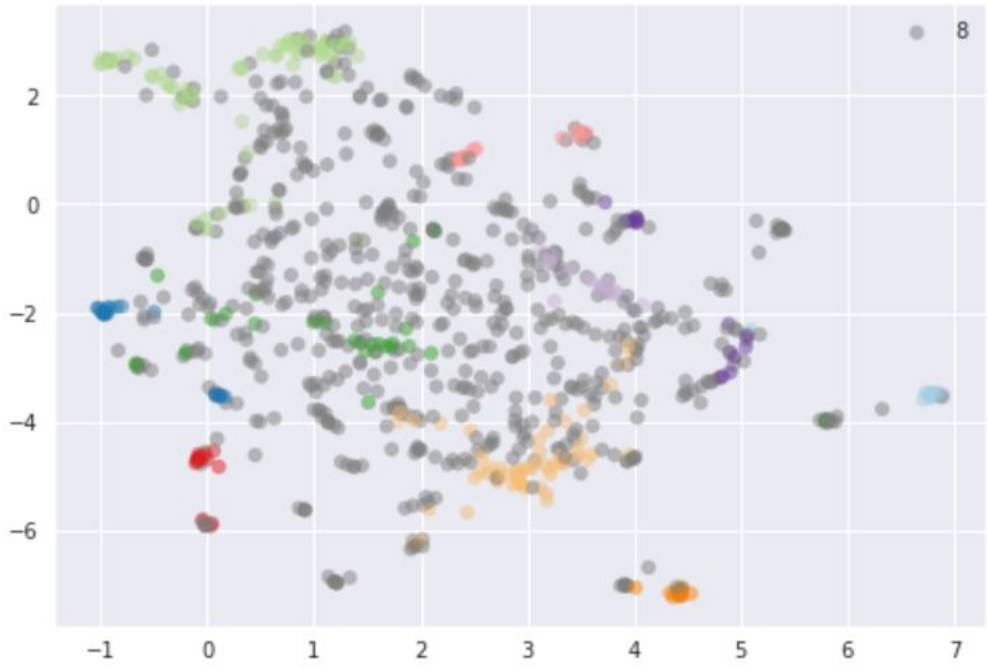
Unsupervised Learning
Though the former efforts have yielded semantically interesting topics within our data set, its representation has been as text. Meaning the topics have been understood semantically, rather than visually. So, an interesting next step could be to visualize the groupings of semantically similar topics. This is the outcome of our unsupervised learning.

To start with, we opted for the most common method of clustering, called K-means. This model finds clusters in text that has been vectorized and reduced, through looking at the k-nearest neighbors of a word. The K-means method requires the input of the number of clusters it should find. Merely inputting a number on gut feeling will however results in meaningless clusters, as the model with find e.g. 500, or 2, clusters if asked.
To approach an estimation of how many clusters to include, we first computed so-called *silhoutte* scores, which are scores that show how much can be explained of the data by a given number of clusters. The resulting plot of those scores yielded an almost straight line, positively correlating the number of clusters with the amount of data explained. There was however a flattening between 15 and 20 clusters, indicating that not much would be gained from surpassing 15 clusters. Ultimately, the conclusion was that our bibliographical data contains many many clusters, and that 15 as an input would be an appropriate trade-off between explainability and interpretability.

The K-means clustering showed 15 clusters that seemed to largely overlap, with the exception of a few more distinct clusters. Most clusters also had low density, which makes it difficult to interpret. To compare its effectiveness with the prior topic classification derived from LSA, we compared cluster by inspecting the most frequent words. The comparison was difficult, perhaps indicating that the success of K-means was not all to great.

Subsequently, we chose to adopt another clustering model, called HDBScan. This model has proved more efficient at handling bibliographical data where many clusters of low volumes exist (Snow, 2018).

We plottet the clusters both using Umap and HDBscan, where Umap helped with dimensionality reduction. The HDBscan plot can be seen here:

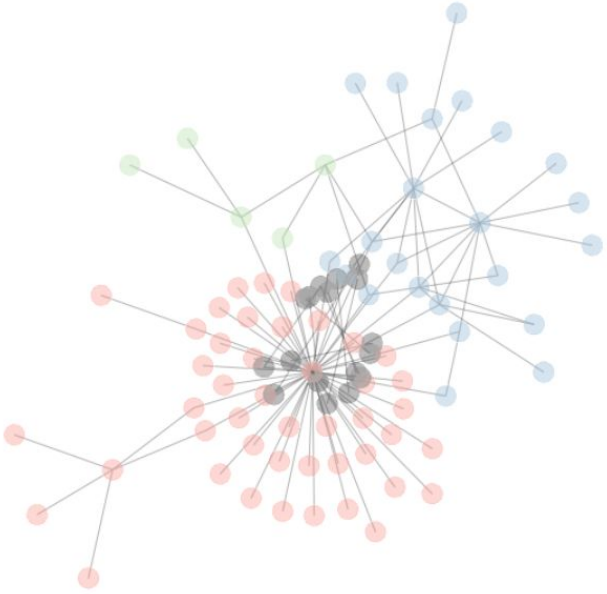


When plotting, we made sure to only color those clusters with abstracts that shared more than 7 words, leaving the rest grayed out. This drastically improved the interpretability of the clusterer.
From these clusters again compared the the most frequent cluster words with those of the LDA. This time, with a noticeable improvement in both in-cluster semantic similarity of words and similarity between the HDScan and LDA clusters.

Network Analysis

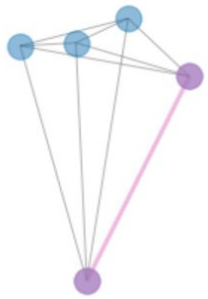
The first part of our network analysis sought to uncover the network between how articles reference each other. Looking at the in degree centralities we found that there is a smoothe change in how many articles reference each other. But looking at the eigenvector centrality we found 3 really important articles, with a far higher eigenvector centrality than the subsequent articles..

The second network we explored was looking at connections between articles that share several words in their titles. Looking at the network between articles there share at least 6 words in their titles we found one article with really 'common' words in its title. We also used an unsupervised machine learning algorithm on the network to find communities. We found 3 communities as seen on the graph below:



Looking at the most central article in each cluster, we found that the first community focused on: Capabilities, adaptability and ambidexterity. The second was about:Export performance, responsiveness and innovativeness. And the third: Structural impact.

The last piece of network analysis we did was to build a collaborator recommender, by finding open triangles in a network between authors. This way we found 512 recommendations, some of them even with multiple neighbours to support the recommendation.



Here we see an example of a recommendation. In the original graph the pink edge did not exist, but seeing how close the two pink nodes are, the two authors would probably be good collaborators.

Conclusion

This M2 project sought to investigate a bibliographical data set of business literature concerning agility. The textual data was preprocessed and used in count-based and unsupervised NLP models, to find valuable trends and clusterings. Unsupervised learning yielded a select number of clusterings based on semantic. As did Network Analysis, but with the additional consideration of how authors might want to collaborate.

References:

Snow, Michael. (2018). *Unsupervised Document Clustering with Cluster Topic Identification.*