Using Formal Concept Analysis for Corpus Visualisation and Relevance Analysis

Fabrice Boissier, Irina Rychkova, Bénédicte Le Grand

Using Formal Concept Analysis for Corpus Visualisation and Relevance Analysis

Fabrice BOISSIER[1][a], Irina RYCHKOVA[2]and Bénédicte LE GRAND[2]
[1]*LRE, EPITA, 14-16 Rue Voltaire, Le Kremlin-Bicêtre, France*
[2]*CRI, Université Paris 1 Panthéon - Sorbonne, 90 Rue de Tolbiac, Paris, France*
fabrice.boissier@epita.fr, {irina.rychkova, benedicte.le-grand}@univ-paris1.fr

Keywords: Corpus Visualization, Relevance Visualization, Topic Modeling, Formal Concept Analysis, Text Mining, Natural Language Processing.

Abstract: Corpora analysis is a common task in digital humanities that profits from the advances in topic modeling and visualization from the computer science and information system fields. Topic modeling is often done using methods from the Latent Dirichlet Allocation (LDA) family, and visualizations usually propose views based on the input documents and topics found. In this paper, we first explore the use of Formal Concept Analysis (FCA) as a replacement for LDA in order to visualize the most important keywords and then the relevance of multiple documents concerning close topics. FCA offers another method for analyzing texts that is not based on probabilities but on the analysis of a lattice and its formal concepts. The main processing pipeline is as follows: first, documents are cleaned using TreeTagger and BabelFy; next, a lattice is built. Following this, the mutual impact is calculated as part of the FCA process. Finally, a force-based graph is generated. The output map is composed of a graph displaying keywords as rings of importance, and documents positioned based on their relevance. Three experiments are presented to evaluate the keywords displayed and how well relevance is evolving on the output map.

1 INTRODUCTION

In the age of data, knowledge is an essential factor that increases the capacity to make the best decisions (North and Kumta, 2018). Data visualization remains a difficult task for knowledge extraction as numerous visualizations are available and each business and/or application face their own challenges (Andrienko et al., 2020)(Padilla et al., 2018)(Engebretsen and Kennedy, 2020). It requires a perfect understanding of the goal to achieve and the manipulated data.

Visualizing text corpora is a cross-domain application of interest for information systems and digital humanities (Jänicke et al., 2015). Improvements in visualization lead to better analysis of multiple types of texts like historical newspapers (Menhour et al., 2023) or even poetry (De Sisto et al., 2024). However, few work has been done to visualize the relevance between documents based on their topics. *Topic modeling* also contributes to texts' analysis as a research orientation of the information retrieval field by "*uncovering latent text topics by modeling word associations*" (Hambarde and Proenca, 2023). Topics are simply a list of words that share a similar theme: either each word is strongly/directly expressing the theme, or the collection of words illustrates an abstract theme by their semantic links (but alone, they would not make sense). Multiple topic modeling methods exist (Alghamdi and Alfalqi, 2015)(Kherwa and Bansal, 2019). They usually consider documents as a *bag-of-words* where the order of words is not important; only their occurrence in each document is important. Recent advances in natural language processing (NLP) also introduced neural networks combined with traditional methods, allowing the capture of the context of words within documents and reusing it to analyze newer documents.

In this paper, we explore the use of *Formal Concept Analysis* (FCA) (Ganter and Wille, 2012) instead of more traditional topic modeling methods, and we propose a visualization of the main keywords of a corpus and documents' relevance on a force-based graph. FCA is known as a viable text mining method (Carpineto and Romano, 2004) and is a good candidate for multiple applications in the knowledge field (Poelmans et al., 2013). FCA has been used in

[a] https://orcid.org/0000-0002-0067-6524

conjunction *with* a topic modeling method (Akhtar et al., 2019) but not *instead* of it. The strength of FCA resides in the fact that it analyzes the relationships within data and produces a lattice that can be used for calculating useful measures like similarities.

The paper is organized as follow: First, we explain the main methods currently used in topic modeling as well as topic and relevance visualizations. Second, we present our processing pipeline from the input documents to the output map. Then, we present three experiments, the text materials used, their topics, and their relevance. Finally, we discuss the results and conclude.

2 RELATED WORKS

2.1 Topic modeling

Analyzing documents implies creating statistics on the used terms. *Term Frequency - Inverse Document Frequency* (TF-IDF) (Salton, 1983) calculates a ratio from the frequency of each term to the total number of documents. Documents are therefore seen as a ratio of words independent of their ordering, like a *bag-of-words*. TF-IDF is not exactly a topic modeling method, but it shows the importance and uniqueness of terms within the corpus.

Latent Semantic Analysis (LSA) (Deerwester et al., 1990) transforms documents into a *latent semantic space* from which multiple outputs can be analyzed. The main method behind LSA is the *singular value decomposition*, which produces three matrices based on a parameter K given by the user: a matrix of *terms per K features*, a matrix of *K features per K features*, and a matrix of *K features per documents*. Multiple analyses can be done on those matrices, but in our case, the *terms per K features* one is the most important as it allows us to know how well each term is linked to each feature (that in fact represents latent topics). One problem induced by LSA is that it can't manage polysemy: each term is used as the same entity in any document. Homonyms, like synonyms and even various forms of the same word, can produce an inconsistency because of the missing context of each document. Standardization of the input, like stemming or even lemmatization, can partially leverage this problem.

Probabilistic Latent Semantic Analysis (PLSA) (Hofmann, 1999) is an upgrade of LSA that introduces a probabilistic point of view by building a generative model for each text corpus. Because topics are scattered within documents, probabilities help find the terms that compose them. PLSA relies on an *aspect model* (the probabilities between terms, documents, and the latent topics) and a *mixture decomposition* that obtains similar results as the singular value decomposition, allowing PLSA also to build the three matrices of LSA.

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is also a generative probabilistic model that aims to model text corpora. It aims to improve the PLSA mixture decomposition by using a hierarchical Bayesian model. LDA allows not only the finding of topics of words within documents, in the case of text corpora, but also its usage with more confidence than PLSA as a probabilistic generative model in multiple domains.

LDA can be combined with *Bidirectional Encoder Representations from Transformers* (BERT) (Devlin et al., 2018) in order to increase the quality of its results (Peinelt et al., 2020)(George and Sumathy, 2023). BERTs are a collection of pre-trained neural networks designed to help researchers in NLP by considering words and their neighbors before and after them. By their nature, BERTs do not produce a list of topics but can be used to generate a summary. Similarly to BERTs, it can be noted that *Generative Pre-trained Transformers* (GPT) (Brown et al., 2020) achieve numerous NLP tasks. Still, they also share the drawback of not explaining the weights within the neural network to build their answers.

Correlated Topic Model (CTM) (Blei and Lafferty, 2006) is derived from LDA and uses another distribution in order to better capture topics and their relations within documents. Indeed, when a document concerns a theme, it usually talks about some neighbor topics: a text about travel might talk about tourism, beaches, and airplanes, but probably not about fighter jets. Topics are uncorrelated in LDA because of the Dirichlet distribution, whereas in CTM, thanks to the logistic normal distribution, topics are correlated and present links to one another. The presence of a topic triggers the possibility of finding one or multiple other topics. CTM also proposes a visualization of topics: each topic is represented in a bubble of words, and each bubble is linked with the other correlated topics.

2.2 Visualization of topics and relevance

The presentation of topics is important as reading a list might confuse a human: a list forces an order of reading, which is not always the best one depending on the context. Visualization of topics is usually made of tag clouds (Singh et al., 2017)(Lee et al., 2010), which is convenient but shows terms as a whole block. Some visualizations like (Singh et al., 2017) or (Gretarsson et al., 2012) show the distinction between topics and the terms composing them. However, if a document used in the corpus is irrelevant or contains too many irrelevant parts, the results might be altered without the user being informed. In addition, each document must be deeply reviewed to find the discrepancy. Few work has been done to explicit the relevance of documents one to another based on their topics.

(Assa et al., 1997) presents a relevance map between topics and keywords in the case of web search queries. Their proposal uses dimension reduction to create a 2D map and gravitational forces for placing nodes. Similarly, VIBE (Olsen et al., 1992)(Olsen et al., 1993) and its variants (Ahn and Brusilovsky, 2009) create *Points of Interest* (POI) that act precisely like topics around which documents are placed based on relevance. The main drawback is that it does not highlight the most important common words and topics but only places resources based on their relevance to all the topics found.

(Fortuna et al., 2005) proposed a map of terms and documents based on *MultiDimensional Scaling* (MDS) methods. Terms and documents are placed on a two-dimensional map, and the background color varies based on the density. The main drawback of this contribution is the difficulty of getting a clear overview of the documents and terms.

In (Newman et al., 2010), the authors proposed a topic map after studying topic modeling and 2D projection. First, they compare three topic modeling methods and end by using LDA. Next, they compare three methods of projection, namely *Principal Component Analysis* (PCA), *Local Linear Embedding* (LLE), and *Force Directed Layout* (FDL) which is the best. The topic map presents documents as nodes colored by their most important topic. Their position depends on their relevance to one another. However, the authors concluded that the evaluation of visualization is complex and could be made only by human judgment; in addition, they also stated that maps with a dozen documents are probably the most accurate and valuable in understandability and navigation for a human.

TopicViz (Eisenstein et al., 2012) proposes a visualization of documents and topics by nodes with a force-directed layout and, more importantly, interaction with the user. The topic modeling method used is LDA. The user can pin topic nodes in the workspace, making the documents float based on their relevance to each topic. Such a map allows one to distinguish which document is more relevant to some or more topics based on its position. The user can also pin document nodes, making the topics float between them. This visualization is particularly interesting because document and topic nodes can be pinned, allowing it to show relevance. However, the user must pin the nodes himself in order to see the relevance. Based on the number of detected topics, deciding where to pin each topic to see the documents' relevance better might be difficult.

PaperViz (di Sciascio et al., 2017) is a dedicated tool for researchers during the paper-gathering step. It offers multiple views for multiple contexts: tree hierarchy for search queries, a tag cloud of the 20 most frequent terms, the strength of the relationship between documents and a search query or a collection, and references management. The main strength of PaperViz is its completeness in the user interface.

3 VISUALIZATION PIPELINE WITH FORMAL CONCEPT ANALYSIS

The visualization pipeline comprises two main phases (see Figure 1): *semantic pre-processing*, where documents are analyzed in order to produce an occurrence matrix of terms per documents, and *structural analysis*, where the matrix is analyzed in order to create a graphical representation of the relevance. Globally, the pipeline relies on natural language processing (NLP) methods in the first phase and formal concept analysis (FCA) methods in the second phase.

3.1 Semantic Pre-Processing

The *semantic pre-processing* phase aims to extract the most important terms and concepts from each document and gather them within a matrix representing the whole text corpus. This phase is composed of 5 steps as follows:

(PI.0) **Selection of documents by the user**: the user selects the documents for analysis and comparison. Two requirements must be fulfilled for the best results: each document must have enough content, and the content must be mainly textual.

(PI.1) **Extraction of texts**: each document is transformed into a regular flat text file. This step relies on optical character recognition (OCR) methods. In our experiments, we used PDFtoText[1] as the OCR.

(PI.2) **Cleaning of extracted texts**: each text is cleaned in order to increase its quality and reduce its size, typically by removing the useless spaces and artifacts that the OCR would have created and, eventually, some of the stop words. In our experiments, we

[1] https://www.xpdfreader.com/pdftotext-man.html

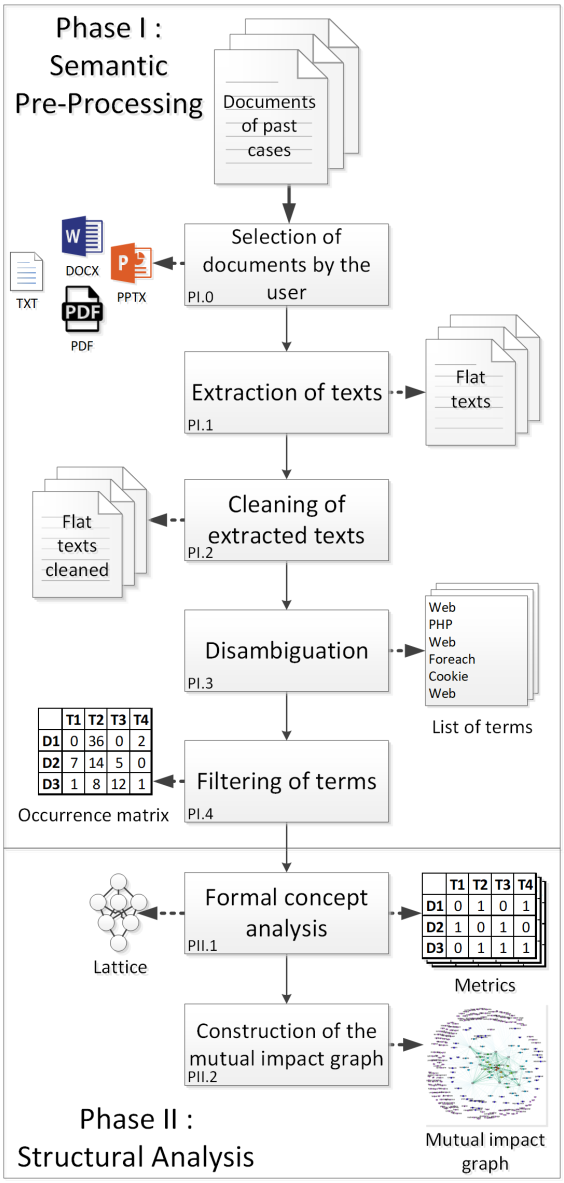

Figure 1: The main steps of the pipeline

used TreeTagger (Schmid, 1994)(Schmid, 1995) with a custom list of words to keep.

(PI.3) **Disambiguation**: each cleaned text is transformed into a list of named entities by resolving polysemy and synonymy problems. Advanced NLP methods are required for this task. In our experiment, we used BabelFy (Moro et al., 2014) as it understands multiple languages and calculates three scores for each recognized named entity. The named entities are also transformed into unique references from BabelNet (Navigli and Ponzetto, 2012), allowing us to manipulate the exact same named entities in all documents, whatever the input languages are.

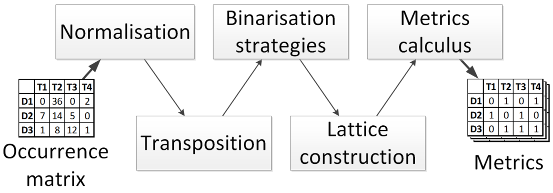

Figure 2: The formal concept analysis (PII.1) sub-steps

(PI.4) **Filtering of terms**: the most irrelevant named entities are removed based on the *coherence score* attributed by BabelFy in the previous step. In our experiments, we require a coherence score of at least 0.05 to keep a named entity. This score was empirically chosen because it removes way more irrelevant named entities than relevant ones.

3.2 Structural Analysis

The *structural analysis* phase comprises two steps that calculate metrics in order to produce the mutual impact graph showing the relevance of documents.

(PII.1) The *formal concept analysis* is the first step, divided into five sub-steps (see Figure 2). Its objective is to produce the mutual impact matrix between terms and documents from the occurrence matrix in order to evaluate the relevance of documents.

- *Normalisation*: occurrences of terms per documents are transformed into proportions in order to reduce the length disproportions between documents. Absolute values are converted into percentages per line. Thus, all documents are treated equally.
- *Transposition*: the matrix is transposed in order to change the point of view from documents characterized by occurrences of terms into terms characterized by their appearances within documents.
- *Binarisation strategy*: building the lattice requires a *formal context*, or, in other words, a binary matrix. Multiple strategies of binarization exist (also called *refinement strategies*) (Jaffal et al., 2015). In our case, we use the most simple one: the *direct strategy* that transforms all values > 0 as 1 and keeps 0 as 0.
- *Lattice construction*: the formal context representing terms within documents is used for building a lattice and its *formal concepts* (Belohlavek, 2008). A formal concept is a node containing objects and attributes (at least one of them). In our case, the objects are terms, and the attributes are the documents.
- *Metric calculus*: the lattice is analyzed, and the *mutual impact* (Jaffal et al., 2015) metric is cal-

culated by comparing the appearances of couples of terms and documents within each formal concept. The mutual impact shows how strong the relationship is between each term and each document. This bond is calculated for each term and each document with the formula:

$$MI(O_i,A_j) = \frac{\textit{formal concepts containing } O_i \text{ and } A_j}{\textit{formal concepts containing } O_i \text{ or } A_j}$$

where O_i represents a term and A_j represents a document. The output is a mutual impact matrix with a value representing the bond between each term and each document.

(PII.2) The *construction of the mutual impact graph* is the final step. Its objective is to visualize on a map the terms and their importance within the corpus, as well as the documents and their relevance. The visualization uses the mutual impact matrix as an adjacency matrix and produces a graph of terms and documents. We used Gephi (Bastian et al., 2009) with the ForceAtlas (first version) spatialisation algorithm. Nodes are moved until they find their optimal position thanks to attractive and repulsive forces. Nodes are repulsing each other, and edges attract nodes based on the values of the edges. Because of the input format, the visualization is a bipartite graph: a set of nodes represents documents, and another set represents terms. As presented in Figure 3, the nodes of documents are colored in grey and are linked to numerous nodes of terms. Unlike, these term nodes are colored based on the number of neighbors (the warmer the color, the more the node of term is linked to different nodes of documents). When focusing on the nodes of terms, we can find terms from every document in the center of the map and terms from fewer documents scattered around (terms, present in only one document, are forming specific groups for each document far from the central set). The central set of terms is, in fact, a global view of the text corpus with its keywords. When focusing on the nodes of documents, we can visually see the relevance of documents within the corpus by checking how close documents are to the central set of terms.

4 EXPERIMENT

4.1 Scenarios of evaluation

A proof of concept (Elliott, 2021) has been realized with a specific scenario to check the pipeline's validity and properties. A prototype has been developed[2] and used in three proof of concept demonstrations. A first case expects to visualize the content of 9 PHP courses. The mutual impact graph is visualized to check the corpus' main keywords and the documents' relevance. A second case introduces a Java course in the corpus. The validity of the mutual impact graph is checked by comparing the first and second cases: the Java course should be the most irrelevant because it is not specialized in PHP. A third case is presented in order to check how correcting a document impacts the results.

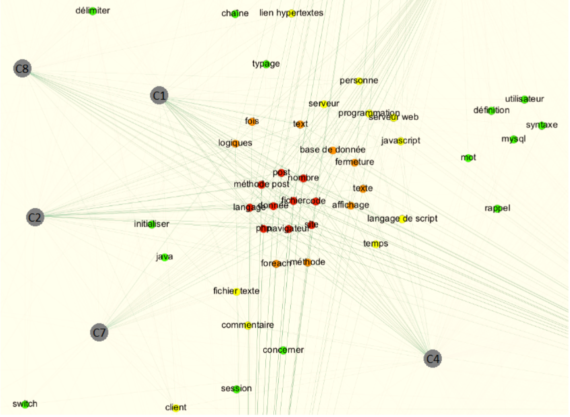

Figure 3: The output map of terms and documents

In the regular case, 9 PHP courses in French are processed through the pipeline. These courses present web development with HTML, PHP, and MySQL for beginners. They are denoted as C1-C9 in the following figures. 6 of these courses are made of slides (C1, C2, C4, C7, C8, C9), and 3 are made of regular texts (C3, C5, C6). A Java course (CJA) in text format is later introduced to check how the mutual impact graph behaves when some errors appear. This course is also in French and in a text format. The third experiment only works on courses in text format. Therefore, we introduced 4 more PHP courses in text format (C11, C15, C17, C19) to avoid disproportion by keeping a close number of input documents as in the other cases.

4.2 General view of the corpus' content

The documents are processed through phase I, and the corpus is transformed into a matrix of occurrences. The mutual impact graph is generated using the *direct strategy* in step PII.1.

Figure 4 shows terms as nodes colored following a cold-warm schema. Red nodes are terms occurring in all the documents, orange nodes are terms occurring in all the documents minus one, etc. In the regular case, the terms in red nodes are: *post*, *nombre* (number), *méthode post* (post method), *fichier* (file), *code*,

[2] https://github.com/metalbobinou/CREA-phd

donnée (data), *langage* (language), *site*, *php*, *navigateur* (browser). These terms are typical of a course on web development with PHP. They are extended with the terms in orange nodes like *base de donnée* (database) or even *foreach*, which are also typical for a website in PHP that uses a database. Terms that are present in fewer documents but still in more than half of the corpus (the yellow and green nodes) are also typical of web development for nearly all of them (*session*, *mysql*, *utilisateur* (user), ...).

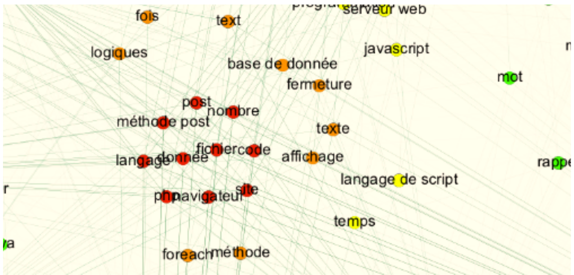

Figure 4: Central set terms of the mutual impact graph in regular case

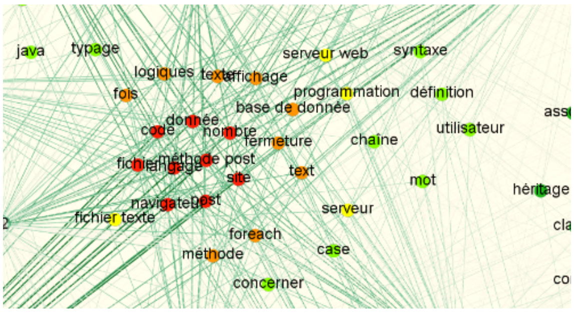

Figure 5: Central set terms of the mutual impact graph in Java case

In Figure 5, the Java course is added to the corpus. The terms in red nodes are the same as in Figure 4, except *php* which becomes an orange node. This behavior is expected as the Java course does not discuss PHP; therefore, one document does not include it. The terms in other colors are still relevant as they mainly concern client-server programming, OOP vocabulary, and similar topics. It can be noted that another color is introduced in order to show that an additional document is present. Nearly all of the terms in green in the first case are now light green. In the entire graph, some of the terms in green are repulsed into dark green nodes (meaning they are missing from one more document). However, a majority of terms from the first case are still present in the second case with the same number of document edges.

4.3 Relevance of documents

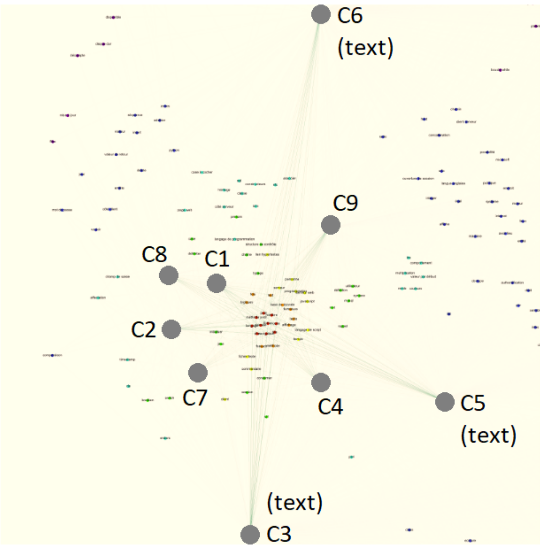

Figure 6: Relevance of documents in regular case

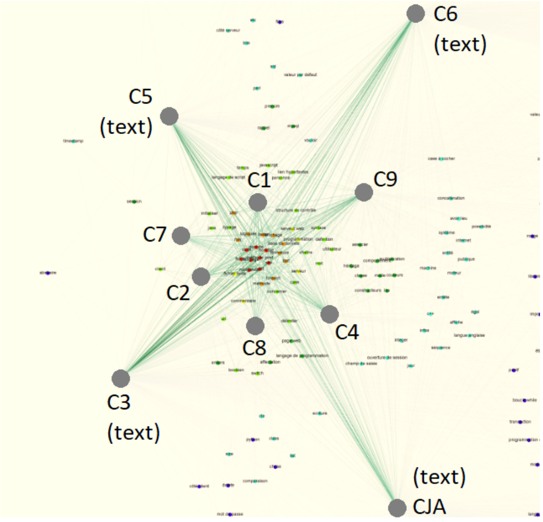

Figure 7: Relevance of documents in Java case

Figures 6 and 7 represent the relevance of documents in two cases: the regular case with only PHP courses, and the Java case with the additional Java course. The relevance's visualization is also produced from the mutual impact graph, except the focus is mainly on the grey nodes representing documents.

In the regular case of PHP courses (Figure 6), the documents in the slide format are closer to the central set than the text ones. In the Java case (figure 7), the Java document (CJA) is the most distant of the central set. It can be noted that C6 is as distant from the central set as CJA. This discrepancy is explained by the fact that the C6 document contains not only a PHP course but also reports of students' projects in more than half of the document. These reports discuss various business problems that require a website (online shoe store, online music store, etc). Therefore, the document is not exactly a pure PHP course like the others.

In order to test how the mutual impact graph reacts when a document is corrected, we compared multiple cases of courses while correcting one of them. Document C6 is written in three parts with nearly the same amount of pages: the regular PHP course, the reports of the students' projects, and an advanced PHP course. For the experiment, we used PHP courses in text format only (in order to avoid the effect of mixing slide and text formats), and we corrected document C6 by removing the students' projects first and, later, the advanced chapters. The experiment was also done twice, with and without the Java course, in order to have a better view of the effect of correction on a corpus with and without an irrelevant course.

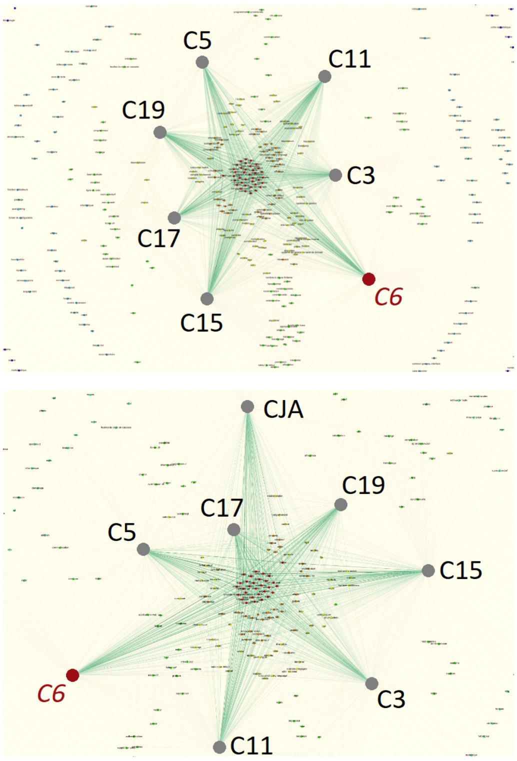

Figure 8: Relevance of text documents (C6 intact)

In the regular case without any modification (Figure 8), C6 is the most outlier in both regular and Java cases. In the Java case, CJA is the sole document nearly as far as C6. After removing the students' projects from C6 (Figure 9), it becomes one of the closest documents from the central set in both

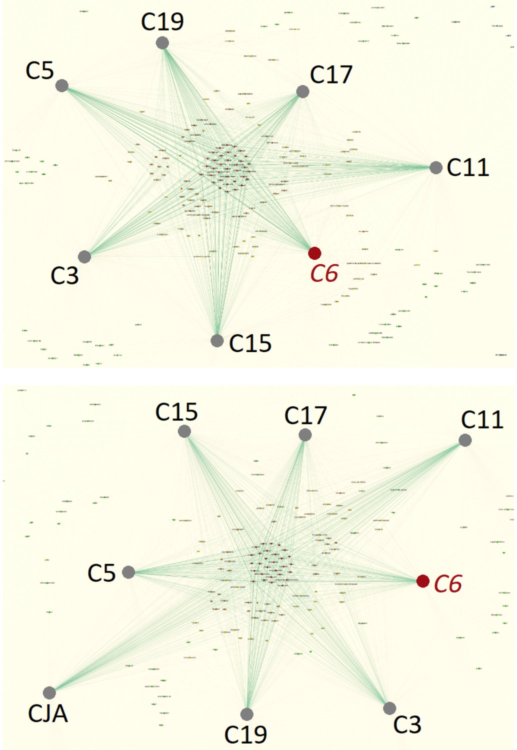

Figure 9: Relevance of text documents (no student projects)

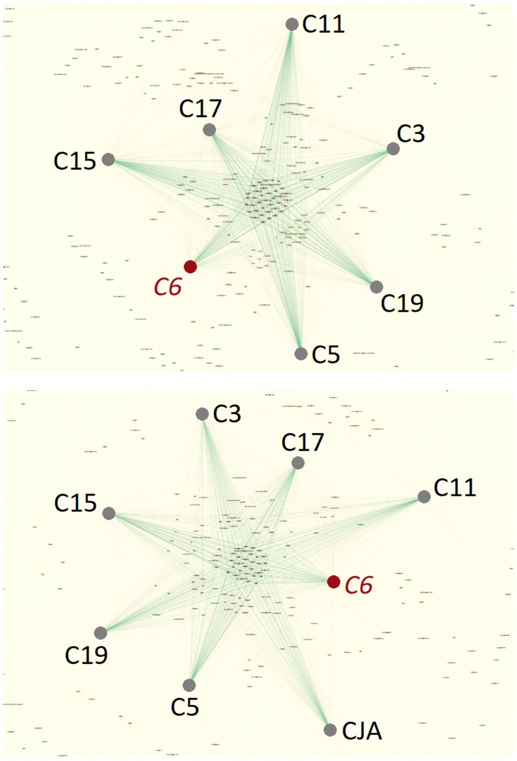

Figure 10: Relevance of text documents in the regular case (no student projects, nor advanced chapters)

cases, indicating that it became way more relevant to the corpus than previously. In the Java case, CJA becomes the most outlier, keeping its irrelevance. Finally, when the advanced chapters are also removed from C6 (Figure 10), it becomes the closest to the central set in the Java case, and the second closest in the regular case. C11 and C15 become the most outliers in the regular case. In the Java case, CJA and C11 are the most outliers.

In each figure from 8 to 10, document C6 becomes closer to the central set thanks to the corrections. It can be noted that CJA stays the most outlier in every case, and the other documents move away a bit. Therefore, the corrections show improvements in the positioning of C6 while keeping the irrelevance of CJA.

5 DISCUSSION

In the mutual impact graph of the regular case (Figure 4), the set of terms present in all of the documents at the center of the graph, shown as red nodes, are relevant to the content of the text corpus, and they also produce a clear summary of the main keywords. Each ring of colors around the central set adds more terms relevant to the corpus. When the Java course is introduced (Figure 5), the terms in all the documents are nearly the same: only *php* is a bit repulsed. As the Java course also mainly talks about programming and partially about web development with dedicated frameworks, the results are nearly unchanged, which is the expected behavior. The mutual impact graph with the terms lets a user quickly get an idea of the subject and keywords composing the corpus. It can be used to quickly discover a new academic field and find the keywords that best describe it. Another usage for this graph would be to help build a book's index: the keywords are highlighted, and the author selects which words to keep or remove.

Concerning the relevance of documents in Figure 7, the Java course is the most distant one with C6 (which contains a lot of unwanted content) in the first experiment. This behavior is perfect for the current case: the teacher who would like to use existing documents is warned that C6 and CJA should be checked more precisely in order to detect if their content is relevant. Even if the text documents are distant from the slide documents, the most irrelevant ones are far away, allowing the user to measure the relevance visually. In the third experiment, the relevance of document C6 is greatly improved because of the corrections applied while keeping the irrelevant document CJA far away. It must be noted that the shape of the graph changes because it reflects the relevance of each document relating to the general relevancy of the whole corpus. The mutual impact graph is a picture of the corpus as a whole: it is not a graph about the relevance of one or some documents against one or some other documents. Multiple use cases can be derived from the mutual impact graph. The graph would help users select the best documents about one or more topics or remove the most irrelevant ones. Another usage would be for a teacher to compare its own course with the existing ones or even with research articles to check how close it is to the state of the art.

The global results show that FCA, with the mutual impact measure, can highlight a corpus' main terms and even show its documents' relevance. It does not create a list of topics nor calculate the probability of each term being included in a topic like LDA, but it does reflect the importance of each term for the whole corpus. This behavior is expected by the nature of FCA: it "*differs from statistical data analysis in that the emphasis is on recognizing and generalizing structural similarities, such as set inclusion relation from the data description, and not on mathematical manipulations of probability distributions*" (Carpineto and Romano, 2004). However, we acknowledge that BabelFy does participate actively in the process of topic modeling by recognizing the named entities and evaluating their relevance to each portion of the text. It must be noted that the construction of the formal context also filters some terms. FCA (formal context's construction + metrics calculus) and BabelFy must be used together, or at least, FCA with an entity linking tool.

6 CONCLUSION

In this paper, we proposed a visualization pipeline for textual corpora analysis based on FCA instead of the usual LDA for the topic modeling step. Mutual impact was used within FCA in order to produce a matrix for the force-based graph algorithm. The pipeline produces a map that can be used in two ways:

- The main keywords are placed by order of importance, allowing the reader to quickly get an idea of the topics contained in the corpus
- Documents are placed based on their relevance to the keywords found, allowing the reader to see an eventual discrepancy in the chosen texts.

The map presents a visualization of the corpus as a whole. Removing a document impacts the visualization because of the absence of a node and because the topic modeling step does not work on the same texts. To evaluate these claims, we presented a case study on multiple PHP courses and an intruder Java course (also about programming). First, a map displayed the most important keywords and the variations with and without the Java course. Then, one of the PHP courses containing more than half of its text about out-of-scope topics was corrected, showing a significant upgrade in the output.

We consider FCA an exciting method for topic modeling and expect to try other metrics on the lattice in order to find more possible usages. Multiple usages and combinations have been already proposed in (Poelmans et al., 2013), but we expect to use the conceptual similarity metric (Jaffal et al., 2015) for an even more precise combination of terms. Also, a deeper comparison with LDA and other newer methods like neural networks might be interesting as the construction of the results does not rely on probabilities and is perfectly transparent thanks to the set theory behind FCA.

REFERENCES

Ahn, J.-w. and Brusilovsky, P. (2009). Adaptive visualization of search results: Bringing user models to visual analytics. *Information Visualization*, 8(3):167–179.

Akhtar, N., Javed, H., and Ahmad, T. (2019). Hierarchical summarization of text documents using topic modeling and formal concept analysis. In *Data Management, Analytics and Innovation: Proceedings of ICDMAI 2018, Volume 2*, pages 21–33. Springer.

Alghamdi, R. and Alfalqi, K. (2015). A survey of topic modeling in text mining. *Int. J. Adv. Comput. Sci. Appl.(IJACSA)*, 6(1).

Andrienko, G., Andrienko, N., Drucker, S. M., Fekete, J.-D., Fisher, D., Idreos, S., Kraska, T., Li, G., Ma, K.-L., Mackinlay, J. D., et al. (2020). Big data visualization and analytics: Future research challenges and emerging applications. *BigVis 2020: Big data visual exploration and analytics*.

Assa, J., Cohen-Or, D., and Milo, T. (1997). Displaying data in multidimensional relevance space with 2d visualization maps. In *Proceedings. Visualization'97 (Cat. No. 97CB36155)*, pages 127–134. IEEE.

Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: an open source software for exploring and manipulating networks. In *Third international AAAI conference on weblogs and social media*.

Belohlavek, R. (2008). Introduction to formal concept analysis. *Palacky University, Department of Computer Science, Olomouc*, 47.

Blei, D. and Lafferty, J. (2006). Correlated topic models. *Advances in neural information processing systems*, 18:147.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Carpineto, C. and Romano, G. (2004). *Concept data analysis: Theory and applications*. John Wiley & Sons.

De Sisto, M., Hernández-Lorenzo, L., De la Rosa, J., Ros, S., and González-Blanco, E. (2024). Understanding poetry using natural language processing tools: a survey. *Digital Scholarship in the Humanities*, page fqae001.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

di Sciascio, C., Mayr, L., and Veas, E. (2017). Exploring and summarizing document colletions with multiple coordinated views. In *Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics*, pages 41–48.

Eisenstein, J., Chau, D. H., Kittur, A., and Xing, E. (2012). Topicviz: Interactive topic exploration in document collections. In *CHI'12 Extended Abstracts on Human Factors in Computing Systems*, pages 2177–2182.

Elliott, S. (2021). Proof of concept research. *Philosophy of Science*, 88(2):258–280.

Engebretsen, M. and Kennedy, H. (2020). *Data visualization in society*. Amsterdam university press.

Fortuna, B., Grobelnik, M., and Mladenic, D. (2005). Visualization of text document corpus. *Informatica*, 29(4).

Ganter, B. and Wille, R. (2012). *Formal concept analysis: mathematical foundations*. Springer Science & Business Media.

George, L. and Sumathy, P. (2023). An integrated clustering and bert framework for improved topic modeling. *International Journal of Information Technology*, 15(4):2187–2195.

Gretarsson, B., O'donovan, J., Bostandjiev, S., Höllerer, T., Asuncion, A., Newman, D., and Smyth, P. (2012). Topicnets: Visual analysis of large text corpora with topic modeling. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(2):1–26.

Hambarde, K. A. and Proenca, H. (2023). Information retrieval: recent advances and beyond. *IEEE Access*.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

Jaffal, A., Le Grand, B., and Kirsch-Pinheiro, M. (2015). Refinement strategies for correlating context and user behavior in pervasive information systems. *Procedia Computer Science*, 52:1040–1046.

Jänicke, S., Franzini, G., Cheema, M. F., and Scheuermann, G. (2015). On close and distant reading in digital humanities: A survey and future challenges. *EuroVis (STARs)*, 2015:83–103.

Kherwa, P. and Bansal, P. (2019). Topic modeling: a comprehensive review. *EAI Endorsed transactions on scalable information systems*, 7(24).

Lee, B., Riche, N. H., Karlson, A. K., and Carpendale, S. (2010). Sparkclouds: Visualizing trends in tag clouds. *IEEE transactions on visualization and computer graphics*, 16(6):1182–1189.

Menhour, H., Şahin, H. B., Sarıkaya, R. N., Aktaş, M., Sağlam, R., Ekinci, E., and Eken, S. (2023). Searchable turkish ocred historical newspaper collection 1928–1942. *Journal of Information Science*, 49(2):335–347.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Newman, D., Baldwin, T., Cavedon, L., Huang, E., Karimi, S., Martinez, D., Scholer, F., and Zobel, J. (2010). Visualizing search results and document collections using topic maps. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(2-3):169–175.

North, K. and Kumta, G. (2018). *Knowledge management: Value creation through organizational learning*. Springer.

Olsen, K. A., Korfhage, R. R., Sochats, K. M., Spring, M. B., and Williams, J. G. (1993). Visualization of a document collection: The vibe system. *Information Processing & Management*, 29(1):69–81.

Olsen, K. A., Williams, J. G., Sochats, K. M., and Hirtle, S. C. (1992). Ideation through visualization: the vibe system. *Multimedia Review*, 3:48–48.

Padilla, L. M., Creem-Regehr, S. H., Hegarty, M., and Stefanucci, J. K. (2018). Decision making with visualizations: a cognitive framework across disciplines. *Cognitive research: principles and implications*, 3:1–25.

Peinelt, N., Nguyen, D., and Liakata, M. (2020). tbert: Topic models and bert joining forces for semantic similarity detection. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 7047–7055.

Poelmans, J., Kuznetsov, S. O., Ignatov, D. I., and Dedene, G. (2013). Formal concept analysis in knowledge processing: A survey on models and techniques. *Expert systems with applications*, 40(16):6601–6623.

Salton, G. (1983). Introduction to modern information retrieval. *McGraw-Hill.*

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *New methods in language processing*, page 154.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to german. In *In Proceedings of the ACL SIGDAT-Workshop*, pages 47–50.

Singh, J., Zerr, S., and Siersdorfer, S. (2017). Structure-aware visualization of text corpora. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, pages 107–116.