Eindhoven University of Technology

MASTER

A machine learning approach to effective evaluation of project planning across construction projects

van de Laar, Britt J.

Award date:
2024

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

EINDHOVEN UNIVERSITY OF TECHNOLOGY

A machine learning approach to effective evaluation of project planning across construction projects

7CC40 Graduation Project 2023/2024 – 40 ects

Construction management and Engineering – Eindhoven University of Technology

14 December 2023

B.J. van de Laar

1244068

Graduation Committee:

Chairman: Pieter Pauwels

1st Supervisor: Ekaterina Petrova

2nd Supervisor: Rik Eshuis

Company Advisor: Sander Winkel

This master thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Integrity.

Table of Contents

Preface 4
Summary – English 5
Summary – Dutch 7
Abstract 9
List of abbreviations 10
List of figures 11
List of tables 12
List of equations 12
List of listings 13
1. Introduction 14
1.1. Background 14
1.2. Problem definition 15
1.3. Thesis objectives 16
1.4. Thesis outline 17
2. State of the Art 18
2.1. Systematic literature review 18
2.1.1. Project planning 24
2.1.2. Artificial Intelligence 25
AI timeline 25
Machine Learning 27
Unsupervised clustering 29
Mean Shift 30
Agglomerative Hierarchical Clustering 32
Natural Language Processing 33
2.1.3. AI based planning 35
2.2. Bibliometric literature review 37
2.2.1. Project Planning 38
2.2.2. AI 40
2.3. Summary of literature review results 41
3. Methodology 42
3.1. Unsupervised ML algorithms 42
3.2. Data preparation 44
3.2.1. Part of Speech 45
3.2.2. Text between brackets and punctuation marks 45

3.2.3. Stemming and Lemmatization 45
3.2.4. Translation of Dutch words 45
3.2.5. Project specific terms 46
3.2.6. Testing data preparation 46
3.3. Naming clusters 46
3.3.1. Manual naming clusters 46
3.3.2. Naming of clusters based on the most prevalent words 47
3.3.3. Naming of clusters based on the most frequent activity 47
3.4. Labelling of new planning activities 47
3.5. Validation of clusters and labelling algorithm 50
3.6. Methodology conclusion 50
4. Results 52
4.1. Data extraction 52
4.2. Multi Criteria Analysis 54
4.3. Testing of DBSCAN and K-means 57
4.3.1. DBSCAN 59
4.3.2. K-means 60
4.3.3. Final choice 64
4.4. Outcome data preparation 65
4.5. Naming of clusters 67
4.6. Labelling algorithm 69
4.7. Result Conclusion 70
5. Validation of results 71
5.1. Highway tender planning 71
5.2. Waterway execution planning 73
5.3. Conclusion on implementation 73
6. Discussion 74
7. Conclusion 76
7.1. Main contributions 76
7.2. Implications for industry and research 78
7.3. Future work 78
References 80
Appendix A: Complete bibliometric literature review 87
Appendix B: Explanation of ID clusters data 96
Appendix C: Final generalized code Clustering 98
Appendix D: Data preparation code 100

Appendix E: Silhouette scores of different data preparation methods 103
Appendix F: Final generalized code Labelling 108
Appendix G: Outcome different preparations methods 110
Appendix H: Code for generating the name of the labels 129
Appendix I: Outcome labels with similarity score 130
Appendix J: Entire path taken in Results chapter 139
Appendix K: Adjusted code data preparation for labelling codes 141

Preface

Dear Reader,

This document marks the final step in my educational career. Just over six years ago, I started my academic journey in the built environment with my bachelor's degree. Initially aspiring to become a structural engineer, I discovered that the project management aspect of the Built Environment resonated more with my interests. During my student time, I am grateful for the new friends I've made and the opportunities that allowed me to grow personally and professionally beyond the regular study program.

Approaching graduation, I aimed to explore the side of my master's that I had not yet dived into, digitalization. While I had a basic grasp of digitalization processes, I had not yet experienced working myself with it. Entering this graduation project, even the installation of Python was a mystery to me. Now, a little over six months later, I not only use it for thesis-related tasks but also for tackling unrelated challenges. For this, I want to thank Count & Cooper, especially the Digital Project Management team, for guiding me in this process. This experience has deepened my understanding of how data positively impacts construction projects. Throughout this graduation project, I gained insights not only from the work done in this thesis but also from the narratives shared within the company about data in the construction sector.

This thesis would not have been possible without the guidance of my supervisors. First, I would like to thank Ekaterina Petrova and Rik Eshuis for their guidance over the past several months. Second, I want to thank Karolin Winter for helping me, especially with the NLP questions. Next, I would like to thank Sander Winkel for my guidance at Count & Cooper, offering valuable insights into code writing and idea discussions. In closing, I also want to thank my friends, particularly those on floor 5, for the delightful moments during the thesis writing journey. And lastly, to the reader, I appreciate you taking the time to read this thesis; I hope you enjoy reading it.

Britt van de Laar

Eindhoven, December 2023

Summary – English

In response to the challenges faced in planning evaluation within the Architecture, Engineering, and Construction (AEC) sector, this thesis introduces a mechanism rooted in unsupervised machine learning. The study focuses on integrating digital technologies in the AEC industry by recognizing the absence of an automated and systematic approach to planning evaluation between complex projects. The overarching goal is to enhance the digitalization of project planning by applying machine learning and artificial intelligence (AI) to cluster and label planning activities.

The literature review consists of both a qualitative and a quantitative methodology. A systematic literature review was used, with the PRISMA framework for the qualitative method and a bibliometric review in VOSviewer for the quantitative method. The systematic review delves into the history of planning a general understanding of AI and combines the two topics by reviewing the generation of planning principles by AI. Within this literature review, an exploration of critical issues, including Project Planning, Project Control, Classification Techniques, and Artificial Intelligence, is conducted. The Machine Learning section of the literature elaborates on the relevance of unsupervised learning, operating without labelled data. It delves into unsupervised clustering using four algorithms: K-Means, Mean-shift, DBSCAN, and Agglomerative Hierarchical Clustering. First, DBSCAN highlights its capability to identify clusters of different shapes, while K-Means highlights its centroid-based approach. The section also explores natural language processing (NLP) challenges and text vectorization techniques for using text within these unsupervised algorithms. The AI-based planning section identifies four clusters of papers: Automatic Planning, Constraint-Based Planning, Standardized Work Breakdown Structures (WBS), and Classification and Machine Learning. Emphasizing AI's role in automating planning processes and enhancing accuracy, these clusters provide an in-depth understanding of AI's impact on project planning. The conclusion highlights the complexities of automated planning in the AEC industry and suggests a phased transition toward digitalization, with semi-automated planning as an intermediary step. The "Automation in planning pyramid" visualizes the envisioned progression from manual to semi-automated and fully automated planning.

Next, the bibliometric analysis is employed as a quantitative approach, revealing keyword co-occurrence clusters in Project Planning and AI. Strong connections between AI, decision support systems, and project planning underscore opportunities for technology-driven improvements in decision-making, resource allocation, and risk reduction. The findings emphasize the interconnected research environment in construction projects and address the need for technology-driven decision-making tools for improved project outcomes.

The methodology chapter addresses the application of AI in the evaluation of complex planning projects, proposing a starting point for a solution to the limitations of existing AI research in project planning. The focus is creating an evaluation tool by generating a database for assessing project planning activities. Within the methodology, the section on unsupervised machine learning (ML) algorithms exists a multi-criteria evaluation of criteria for clustering algorithms. The two highest-scoring algorithms in terms of scalability, interpretability, and adaptability to diverse cluster shapes and sizes, laying the foundation for the algorithm selection, are tested on a sample data set. The Natural Language Processing (NLP) section elaborates on how to form clustering processes with text—introducing data preparation steps to enhance cluster similarity. This section ensures a coherent presentation of planning activities, from identifying Parts of Speech (PoS) to removing text elements, stemming, lemmatization, and excluding non-English and project-specific terms. The introduction of a second Python code for labelling new activities based on clusters signifies a shift towards the end product of this research. The validation process, executed on two projects, uses this algorithm. It shows the accuracy of the code in new planning scenarios.

In the results chapter, the decisions discussed within the methodological framework translate into outcomes, offering insights into the practicality of the envisioned machine learning algorithm. Following the exploration of the ML algorithms, a multi-criteria evaluation positions DBSCAN and K-Means as the highest-scoring contenders. After testing the algorithms with a sample dataset, the conclusion was to use K-means for the remainder of the research. The chapter continues with a focus on data presentation to the algorithm, acknowledging the critical role of data coherence. The results chapter discusses several data preparation possibilities, such as tokenizing, lemmatizing, stemming, and removing project-specific words. Six different NLP techniques are used to uniformize the data as much as possible. With this decision, the final clustering is made. This final clustering, made with the K-means algorithm, presents numbers-based clusters. However, these numerical cluster names are transformed into textual names to make them more useful in practice. This process gives each cluster the name of the most frequently used activity. However, this process unveils challenges linked to low coherence within specific clusters. Despite suboptimal similarity within clusters, the validation describes the testing of labelling new planning activities within the established clusters. The findings reveal a dependence on clusters with low similarity as primary labels for new activities. Next, the labelling predominantly relies on verbs within activity names for coherence. Challenges stemming from the translation process highlight the need for enhanced cluster similarity to ensure accurate labelling.

In conclusion, this thesis represents a step in exploring the digitization of project planning in the AEC sector using unsupervised machine learning and AI. The insights gained provide valuable perspectives on algorithm selection, data preparation, and labelling challenges in complex projects. The proposed evaluation tool sets the stage for future developments in automated planning processes, emphasizing the potential for technology-driven improvements in planning evaluation between complex projects. Furthermore, the identified challenges described in this report present opportunities for further research. Particularly in the exploration of other fields within AI and ML. These explorations could enhance clustering accuracy and overall efficacy, offering a foundation for continued investigation of methodologies with the potential further to digitalize the project management aspects of the AEC industry.

Summary – Dutch

Als antwoord op de uitdagingen die zich voordoen bij het evalueren van planningen binnen de Architectuur, Techniek en Bouw (AEC) sector, introduceert dit afstudeeronderzoek een mechanisme dat is geworteld in machine learning zonder supervisie. Het onderzoek richt zich op het integreren van digitale technologieën in de AEC-industrie door het erkennen van de afwezigheid van een geautomatiseerde en systematische aanpak voor planningsevaluatie tussen complexe projecten. Het overkoepelende doel is om de digitalisering van projectplanning te verbeteren door machine learning en kunstmatige intelligentie (AI) toe te passen om planningsactiviteiten te clusteren en te labelen.

Het literatuuronderzoek bestaat uit zowel een kwalitatieve als een kwantitatieve methode. Er is een systematisch literatuuronderzoek gebruikt, met het PRISMA-raamwerk voor de kwalitatieve methode en een bibliometrisch onderzoek in VOSviewer voor de kwantitatieve methode. De systematische review gaat in op de geschiedenis van planning en een algemeen begrip van AI en combineert de twee onderwerpen door de vorming van planningsprincipes door AI te onderzoeken. Binnen dit literatuuronderzoek wordt een analyse gemaakt van cruciale onderwerpen, waaronder projectplanning, projectbeheersing, classificatietechnieken en kunstmatige intelligentie. De sectie Machine Learning van de literatuur gaat dieper in op de relevantie van unsupervised learning, wat werkt zonder gelabelde data. Er wordt ingegaan op clusteren zonder toezicht met behulp van vier algoritmen: K-Means, Mean-shift, DBSCAN en Agglomerative Hierarchical Clustering. Allereerst belicht DBSCAN zijn vermogen om clusters van verschillende vormen te identificeren, terwijl K-Means zijn op centroïde gebaseerde benadering belicht. Het hoofdstuk verkent ook de uitdagingen op het gebied van natuurlijke taalverwerking (NLP) en tekstvectorisatietechnieken voor het gebruik van tekst binnen deze algoritmen zonder supervisie. De sectie over AI-gebaseerde planning identificeert vier clusters van papers: Automatisch plannen, op beperkingen gebaseerde planning, gestandaardiseerde work breakdown structures (WBS) en classificatie en machinaal leren. Deze categorieën benadrukken de rol van AI bij het automatiseren van planningsprocessen en het verbeteren van de nauwkeurigheid en geven een diepgaand inzicht in de invloed van AI op projectplanning. De conclusie benadrukt de complexiteit van geautomatiseerde planning in de AEC-industrie en stelt een gefaseerde overgang naar digitalisering voor, met semi-geautomatiseerde planning als tussenstap. De "Automatisering in planning piramide" visualiseert de beoogde progressie van handmatige naar semi-geautomatiseerde en volledig geautomatiseerde planning.

Vervolgens wordt de bibliometrische analyse gebruikt als kwantitatieve benadering, waarbij clusters van trefwoorden in Projectplanning en AI worden onthuld. Sterke verbanden tussen AI, besluitondersteuningssytemen en projectplanning onderstrepen mogelijkheden voor technologiegedreven verbeteringen in besluitvorming, toewijzing van middelen en risicovermindering. De bevindingen benadrukken de onderling verbonden onderzoeksomgeving in bouwprojecten en behandelen de behoefte aan technologiegedreven besluitvormingstools voor betere projectresultaten.

Het hoofdstuk over methodologie behandelt de toepassing van AI bij de evaluatie van complexe planningsprojecten, waarbij een uitgangspunt wordt voorgesteld voor een oplossing voor de beperkingen van bestaand AI-onderzoek op het gebied van projectplanning. De focus ligt op het creëren van een evaluatie-instrument door het genereren van een database voor het beoordelen van projectplanningsactiviteiten. Binnen de methodologie bestaat het gedeelte over algoritmen voor ongesuperviseerd machinaal leren (ML) uit een multicriteria-evaluatie van criteria voor clusteralgoritmen. De twee hoogst scorende algoritmen in termen van schaalbaarheid, interpreteerbaarheid en aanpasbaarheid aan diverse clustervormen en -groottes, die de basis vormen voor de algoritmeselectie, worden getest op een voorbeeldgegevensset. Het gedeelte over de verwerking van Natural Language Processing (NLP) gaat dieper in op de manier waarop

clusteringprocessen met tekst kunnen worden gevormd, waarbij stappen worden geïntroduceerd voor de voorbereiding van gegevens om de overeenkomst tussen clusters te verbeteren. Dit gedeelte zorgt voor een samenhangende presentatie van planningsactiviteiten, van het identificeren van Parts of Speech (PoS) tot het verwijderen van tekstelementen, stemming, lemmatisering en het uitsluiten van niet-Engelse en projectspecifieke termen. De introductie van een tweede Python-code voor het labelen van nieuwe activiteiten op basis van clusters betekent een verschuiving naar het eindproduct van dit onderzoek. Het validatieproces, uitgevoerd op twee projecten, maakt gebruik van dit algoritme. Het toont de nauwkeurigheid van de code in nieuwe planningscenario's.

In het resultatenhoofdstuk worden de beslissingen die binnen het methodologische kader zijn besproken, vertaald naar resultaten die inzicht bieden in de bruikbaarheid van het beoogde algoritme voor machine learning. Na de verkenning van de ML-algoritmen positioneert een evaluatie op basis van meerdere criteria DBSCAN en K-Means als de hoogst scorende kanshebbers. Na het testen van de algoritmen met een voorbeelddataset werd geconcludeerd om K-means te gebruiken voor de rest van het onderzoek. Het hoofdstuk gaat verder met een focus op de presentatie van gegevens aan het algoritme, waarbij de cruciale rol van gegevenscoherentie wordt erkend. Het resultatenhoofdstuk bespreekt verschillende datavoorbereidingsmogelijkheden, zoals tokenen, lemmatiseren, stammen en het verwijderen van projectspecifieke woorden. Zes verschillende NLP-technieken worden gebruikt om de data zoveel mogelijk te uniformeren. Met deze beslissing wordt de uiteindelijke clustering gemaakt. Deze uiteindelijke clustering, gemaakt met het K-means algoritme, presenteert op nummers gebaseerde clusters. Deze numerieke clusternamen worden echter omgezet in tekstuele namen om ze bruikbaarder te maken in de praktijk. Dit proces geeft elk cluster de naam van de meest gebruikte activiteit. Dit proces brengt echter uitdagingen aan het licht die te maken hebben met een lage samenhang binnen specifieke clusters. Ondanks de suboptimale overeenkomst binnen clusters, beschrijft de validatie het testen van het labelen van nieuwe planningsactiviteiten binnen de vastgestelde clusters. De bevindingen onthullen een afhankelijkheid van clusters met een lage overeenkomst als primaire labels voor nieuwe activiteiten. Vervolgens is de labeling voornamelijk gebaseerd op werkwoorden binnen de namen van activiteiten voor coherentie. Uitdagingen die voortkomen uit het vertaalproces benadrukken de noodzaak van verbeterde clustergelijkenis om accurate labeling te garanderen.

Als conclusie kan worden gesteld dat dit afstudeeronderzoek een stap is in het verkennen van de digitalisering van projectplanning in de AEC-sector met behulp van unsupervised machine learning en AI. De verkregen inzichten bieden waardevolle perspectieven op algoritmeselectie, datavoorbereiding en labeluitdagingen in complexe projecten. De voorgestelde evaluatietool vormt de basis voor toekomstige ontwikkelingen in geautomatiseerde planningsprocessen en benadrukt het potentieel voor technologiegedreven verbeteringen in planningsbeoordeling tussen complexe projecten. Bovendien bieden de in dit rapport beschreven uitdagingen mogelijkheden voor verder onderzoek. Met name in de verkenning van andere gebieden binnen AI en ML. Deze verkenningen zouden de nauwkeurigheid van de clustering en de algehele doeltreffendheid kunnen verbeteren en bieden een basis voor verder onderzoek naar methodologieën met het potentieel om de projectmanagementaspecten van de AEC-industrie verder te digitaliseren.

Abstract

This thesis addresses the challenges of planning evaluation in the Architecture, Engineering and Construction (AEC) sector by developing a mechanism for clustering and labelling planning activities. The research objectives include the development of a method for clustering planning activities using unsupervised machine learning and implementing a labelling system for new activities based on these clusters. Real project applications validate the mechanism's effectiveness, focusing on identifying bottlenecks and optimizing resource allocation. The central problem tackled in this thesis is the lack of an automatic and systematic approach to planning evaluation, particularly concerning historical data, lead times, and the absence of automation and intelligent computing in planning processes. Addressing this issue, the thesis explores current planning principles, generic category clustering, category definition, and the requirements for clustering and labelling activities. The thesis is structured with a systematic and bibliometric literature review, an explanation of the research methodology, presentation of results, validation of clusters and labels, and a discussion of findings and potential future research directions. This research contributes to bridging the gap between traditional and automated planning methods in the industry, taking a significant step towards the automated evaluation of project planning within the AEC sector.

Keywords: *Project planning evaluation, Automated labelling, K-means clustering, Artificial Intelligence, Unsupervised Machine Learning*

List of abbreviations

AEC	Architecture Engineering and Construction
AGI	Artificial General Intelligence
AI	Artificial Intelligence
ANI	Artificial Narrow Intelligence
ASI	Artificial Strong Intelligence
BIM	Building Information Modelling
BoW	Bag-of-Words
CAD	Computer-Aided Design
CDE	Common Data Environment
CPM	Critical Path Method
CSV	Comma-Separated Values
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DES	Discrete-Event Simulation
DTM	Document-Term Matrix
MCA	Multi-Criteria Analysis
ML	Machine Learning
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
OHE	One-Hot Encoding
PERT	Program Evaluation and Review Technique
PMBOK	Project Management Body of Knowledge
PMI	Project Management Institute
POS	Part of Speech
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
SQL	Structured Query Language
SVD	Singular Value Decomposition
TF-IDF	Term Frequency-Inverse Document Frequency
WBS	Work Breakdown Structure
WP	Work Package

List of figures

Figure 1 Paper selection process PRISMA 21
Figure 2 Types of AI 26
Figure 3 Types of Machine Learning (Oladipupo, 2010) 27
Figure 4 K-means clustering visualization 29
Figure 5 Mean-shift visualization (Sippel et al., 2015) 30
Figure 6 DBSCAN parameters visualisation (Pandey, 2020) 31
Figure 7 Hierarchical clustering visualization (Nielsen, 2016) 32
Figure 8 Automation in planning pyramid 37
Figure 9 Bibliometric map of Keywords of Project Planning articles based on occurrences 38
Figure 10 Bibliometric map of Keywords of AI articles based on occurrences 40
Figure 11 General steps in method of algorithms 43
Figure 12 Explanation of the clusters and labelling process 48
Figure 13 Labelling methodology 49
Figure 14 Complete overview of methodology 51
Figure 15 Method DBSCAN and K-Means 58
Figure 16 Outcome DBSCAN clustering 60
Figure 17 Silhouette score clusters sample data 62
Figure 18 Outcome K-means clustering sample data 63
Figure 19 Visualization choice of method 64
Figure 20 Method for the assessment of the clustering results 65
Figure 21 Silhouette score of final clustering 66
Figure 22 Zoomed in Silhouette score of final clustering 66
Figure 23 Boxplot of distribution of the percentages of all clusters 68
Figure 24 Decision path Results 70
Figure 25 Comparison of used clusters A15 with avarage of all clusters 72
Figure 26 Bibliometric map of Keywords of Project Planning articles based on occurrences 88
Figure 27 Bibliometric map of Keywords of Artificial Intelligence articles based on occurrences 92
Figure 28 Boxplot of similarity score of original data 119
Figure 29 Distribution of activities 119

List of tables

Table 1 PRISMA search concepts and synonyms 20
Table 2 List of literature with corresponding themes 22
Table 3 Overview algorithms and the corresponding category (Amasha et al., 2020) 29
Table 4 Planning innovations within read papers 36
Table 5 Selection of data project Veenix 53
Table 6 MCA on unsupervised clustering technique 54
Table 7 Part of clusters and labels for further explanation 67
Table 8 Original names of the activities of the A15 planning 71
Table 9 Adjusted names of the activities of the A15 planning after data preparation 72
Table 10 Clustering on noun "Sand" 72
Table 11 Project Planning keywords map with occurrence per keyword 88
Table 12 Artificial Intelligence keywords map with occurrence per keyword 93
Table 13 Overview cosine similarity and silhouette score 112

List of equations

Equation 1 Systematic literature search query 20
Equation 2 Validation search query 21

List of listings

Listing 1 SQL query to retrieve data 52
Listing 2 DBSCAN algorithm testing 60
Listing 3 Sample K-means clustering till silhouette score 61
Listing 4 Sample K-means clustering after silhouette score 63
Listing 5 Trained model K-Means 69
Listing 6 Labelling code 70
Listing 7 Final generalized code Clustering 99
Listing 8 identify PoS and extracting nouns and verbs 100
Listing 9 removal of punctuation marks and text between brackets 100
Listing 10 Stemming of words 100
Listing 11 Lemmatization of words 101
Listing 12 Removal non-English words 101
Listing 13 Categories of removal of project specific terms 102
Listing 14 Removal residual abbreviations and word 102
Listing 15 Final generalized code Labelling 109
Listing 16 Adjusted code for A15 141
Listing 17 Adjusted code for Twente Canals 141

1. Introduction

The introduction of this thesis will cover the background of the research. Next to this, the problem statement and the research relevance are defined. This chapter will end with covering the topics discussed in the remainder of this thesis.

1.1. Background

The Architecture, Engineering, and Construction (AEC) industry deals with challenges that call for innovative solutions. One of these challenges is the industry's slow adoption of digital technologies in project management. According to a report by McKinsey & Company (2020), the construction industry is one of the least digitized sectors, lagging behind the agriculture industry. Historically, the industry has relied on traditional methods of project management, which caused the sector not to benefit from adopting existing digital technologies (Sawhney et al., 2020). However, it is now widely acknowledged that these digital technologies can potentially boost the AEC industry (Manzoor et al., 2021). This need for digitalization has already accelerated in the last 30 years. When the COVID-19 pandemic hit, the more prominent need for digitalization due to remote work became apparent. The McKinsey report indicates this pandemic's influence on remote work and digitalization. McKinsey's report also suggests that this forced digitalization could further accelerate the adoption of digital technologies within the industry. The driver for the AEC industry to embrace digitalization lies in enhancing efficiency, productivity, and quality and curbing failure costs, errors, and manual efforts (Hosseini & Chileshe, 2014; Howard et al., 1989; Roberts et al., 2018). Specifically, focusing on digital tools during the planning phase can empower project managers with more insights into project plan viability and feasibility, aiding in evaluating different planning scenarios.

The issue at hand revolves around the insufficient digitalization in project management within the AEC industry, particularly concerning project planning. Building Information Modelling (BIM) is crucial when addressing the AEC sector's digital transformation. In the BIM Handbook (Sacks et al., 2018), the evolution of data and information management in construction projects is traced from a document-centric approach to a more streamlined, digital, and model-based methodology. The journey began with the advent of 3D modelling in the 1970s, rooted in the advancements of Computer Aided Design (CAD) technology. Charles Eastman's pivotal 1975 article, "The Use of Computers instead of Drawings in Building Design," is worth noting in this context. Eastman proposed a transformative vision, advocating the replacement of traditional drawings with computers, embracing a database-oriented approach, and emphasizing visual and quantitative analyses. This conceptualization laid the foundation for what later is recognized as Building Information Modelling (BIM).

Contrary to the notion that these developments were synonymous, they marked distinct trajectories. The introduction of the Building Description System in 1975 triggered two divergent research tracks: Building Product Models in the USA and Product Information Models in Europe (Sacks et al., 2018). The primary focus was not solely on software development but rather on reshaping data handling methodologies and transforming the practices of AEC professionals. This divergence led to the introduction of Archicad in 1984 and the inaugural version of Revit in 1999, each playing a pivotal role in this evolving landscape. Notably, while CAD also originated in the 80s, its evolution from 2D to 3D unfolded somewhat concurrently with the introduction of 3DS Max in the 90s and Sketchup in 2000—these milestones collectively illuminate technology's intricate evolution and integration in reshaping project management within the AEC industry.

Even though digitalization with BIM is a step in the right direction, some aspects of the AEC industry still lack this digitalization. One of these aspects is the project planning process, which is critical to achieving the goals and objectives of a construction project. Planning is essential to ensuring a

construction project's success, and knowing if the planning is realistically made is necessary. In project planning, activities are scheduled with a specific timestamp, although the actual lead time may differ from the anticipated length. This inconsistency can be explained by various factors, including unexpected delays, e.g. labour shortage and unrealistic project scheduling (Mbala et al., 2019). To better understand the relationship between the planned and actual duration times, it is necessary to analyse planning data from different projects. Even if other projects have similar activities, comparing these projects can be difficult because the activities in these projects are not identical. An overall overview is complex to achieve when comparing different projects with each other due to the large amount of data. To create this coherence, digitalization and automatization can play a significant role. Therefore, this research will concentrate on digitalising the evaluation process between the planning of different projects. This digitalization will contribute to achieving a first step in the digitalization of planning, focusing on evaluating various complex projects. In this thesis, large-scale infrastructure projects are used to accomplish this.

1.2. Problem definition

The research problem that this thesis aims to address is the difficulty of planning evaluation based on historical planning and lead time in the construction industry. There is currently no systematic way to compare different plans and determine whether the activities in a plan take longer than average. This is a significant limitation of the construction industry's current planning principles.

Additionally, the present construction industry planning processes lack automation and intelligent computing, limiting their efficiency in complex projects. As a result, an innovative solution that uses automation and intelligent computing to analyse planning based on historical data and lead time systematically is required. Technologies such as automation and intelligent computing, potentially integrating (un)supervised AI approaches or Natural Language Processing (NLP), are explored to move to a more automated approach for planning evaluation. These tools promise to systematically analyse planning procedures based on historical data and lead time, presenting an opportunity to revolutionize project management. Therefore, this thesis aims to answer the following main research question:

"How can machine learning contribute to the effective evaluation of project planning across different construction projects?"

Several sub-questions have been defined to answer this main research question. These sub-questions aim to serve as a guide to answer the main question. The sub-questions are the different steps that will answer the main research question.

1. *"What is the current state-of-the-art knowledge of planning principles, artificial intelligence and the implementation of Artificial Intelligence into planning?"*
2. *How can machine learning techniques cluster planning activities based on task similarity?"*
3. *"How can planning activities of the same type be successfully clustered together?"*
4. *"What are the essential criteria and requirements for accurately labelling new planning activities and assigning them to clusters with similar activities?"*
5. *"To what extent does implementing the clustering and labelling mechanism work efficiently over multiple projects?"*

1.3. Thesis objectives

As stated in the main research question, the primary objective of this research is to evaluate project planning effectively across different construction projects. Therefore, this research emphasises the need to evaluate planning activities between construction projects. Since these activities are the building stones of planning, they will be the central part of this research. When evaluating and comparing these activities, knowing which are the same type is essential. There is not yet a systematic approach to define these types of activities. Therefore, this thesis will focus on finding a way to identify these types of activities systematically. This approach seeks to overcome the limitations of existing practices by introducing a mechanism that employs advanced technologies, such as ML. The specific objectives of this thesis can be summarized as follows:

1. Develop a method for identifying the different planning activities based on relevant criteria or attributes, such as task similarity, resource requirements, or time dependencies. This objective will be called the clustering process.
2. Ensure that similar activities from a new project planning are assigned to the appropriate categories in the first objective. This objective will be called the labelling process.
3. Evaluate the effectiveness of the clustering and labelling process by applying it to actual construction projects, and assess how this process works and if it is a suitable solution for the research problem.

The ultimate goal is to bridge the gap in the AEC industry's current planning methods by introducing an initial approach that innovates the planning evaluation process. This approach provides insights into planning activities, paving the way for a more gradual shift towards automated planning methods in the AEC industry. The proposed mechanism should be a helpful tool for planners without replacing their role, offering a stepping stone toward more advanced automation in the future.

1.4. Thesis outline

This master thesis covers the following components to answer the research questions presented in Section 1.2. The initial step of this research is conducting a literature review, which can be found in Chapter 2. This literature review consists of two parts. First, a systematic literature review is performed, which will cover the state-of-the-art regarding planning innovations (e.g. automatic and constraint-based planning) and digital innovations (e.g. Artificial Intelligence (AI) and Machine Learning (ML)). Second, a bibliographic literature review is conducted. This literature method is used as a validation tool to see if the complete field of the systematic literature review query is explored. The two literature review methods are complemented by other relevant literature sources regarding the background information of several aspects of this thesis (e.g. specific planning tools and historical timelines). The entire literature review will answer sub-questions 1 and 2. Going in-depth on the state-of-the-art regarding the planning and digital innovations ensures the clustering and labelling of data. This is done by looking at past practices regarding this topic and the field of AI for possible answers. Furthermore, the literature also covers the different approaches to classifying the generic categories by diving into the different forms of artificial intelligence and machine learning to generate these clusters.

Chapter 3 of this thesis explains the methodology used in this research. This chapter will investigate the method used to answer sub-questions 3 and 4. This chapter will cover the process of generating the generic categories and the generation of the labelling algorithm. First, the possible unsupervised clustering methods are described. Using a Multi-Criteria Analysis, the two most suitable methods are elaborated on in more detail. These two methods are both tested on a test dataset. This testing of the two methods will determine which will be used for the remainder of this thesis. The choice of text vectorization is also explained since this thesis will use textual data that is converted to numerical data. Next to this is an explanation of the used data and the extraction of it. The method chapter will continue by explaining the clustering and labelling algorithm and the data improvement methods used. The chapter ends with an explanation of the clustering assessment.

The outcome of the previously mentioned clustering results assessment method is discussed in Chapter 4. Results. This chapter will first focus on the results generated by the different data preparation methods. Therefore, this chapter also answers sub-questions 3 and 4. The results of the method with the highest clustering score are explained. This chapter will include the final number of clusters as well as the naming of all the labels corresponding to those clusters.

In Chapter 5, the previously made clusters and corresponding labels will be validated. In this chapter, a new planning is inserted into the labelling algorithm. The outcome of this process will show if the corresponding labels are assigned correctly to the newly imported activities. This chapter will also demonstrate and reflect on the outcome of subquestion 5.

The last part of this thesis is the conclusion. This chapter aims to reflect on the answers to the defined research questions. This chapter will reflect on the decisions made in this research and the results that were found. It will incorporate a critical review of whether the aim of this research is met, the implications of the results on research and practice, and what further research could be done regarding this topic.

2. State of the Art

This thesis incorporates two main components in its literature review:

1. A qualitative literature review conducted through a systematic approach;
2. A quantitative literature review conducted through a bibliometric analysis.

The decision to present the systematic literature review, followed by the bibliometric literature review in this report, is based on several considerations. First, it follows a logical sequence of information presentation, enabling readers to understand the research subject comprehensively. The systematic literature review provides an overview of recent information, identifies key themes, and highlights relevant findings from available studies (Tranfield et al., 2003). This approach is beneficial as it allows readers to familiarize themselves with the scope and depth of the research subject, facilitating their comprehension of the subsequent results from the bibliometric analysis. Moreover, by introducing the systematic review first, the report ensures that the bibliometric analysis is appropriately contextualized within a broader framework. Since using the same search query for both reviews, the bibliometric literature review offers an additional perspective to complement the qualitative insights derived from the systematic review (Zupic & Čater, 2015). The systematic literature review and the bibliometric analysis prove valuable as it offers an in-depth understanding of the research subject, integrating both the qualitative and quantitative dimensions.

2.1. Systematic literature review

The first step in this literature research entails conducting a systematic literature review. This approach aims to provide a comprehensive and structured synthesis of the available literature related to the research topic. A systematic methodology is employed during a systematic literature review to identify, select, evaluate, and analyse relevant studies (Boland et al., 2017). Boland et al. also explain that this process involves searching different databases, defining specific inclusion and exclusion criteria, and critically examining the selected literature. Furthermore, by following the systematic review methodology, transparency, replicability, and minimization of biases in the selection and interpretation of the included studies are ensured. The systematic literature review aims to analyse a wide range of existing literature. In the end, the systematic literature review provides a detailed overview of the current state of knowledge on the research topic and provides a comprehensive understanding of the research subject. The systematic literature review uses the PRISMA framework. PRISMA stands for "Preferred Reporting Items for Systematic Reviews and Meta-Analyses" (PRISMA, n.d.).

Table 1 illustrates the key components that should be covered in the literature review, focusing on the main concepts of Analysis, Project planning, and AI. It also includes the synonyms and how the search queries document the words. This table results in a search query that eventually will generate a list of papers.

Table 1 PRISMA search concepts and synonyms

Concept	Analysing	Project planning	Artificial intelligence
synonym	Analy*	“Project planning*”	“Artificial intelligence”
	Compar*	“Planning activit*”	AI
	Evaluat*		Automat*
	Similar*		“Intelligent computing”
	Categori*		

First, the selected papers must be relevant for reviewing the current state-of-the-art knowledge, which implies that these papers should not be outdated. Meaning, that an exclusion criterion has been set to exclude papers published prior to the year 2016. This will leave the papers published over a roughly five-year time period. Next, papers newer than 2023 are also excluded since that year has not yet been reached. Furthermore, papers published after 2023 are also excluded, as this year has not been reached yet. Secondly, only the literature for which the complete paper is accessible will be included. If a paper is unavailable within the database, it will not be considered for inclusion in this systematic literature review. Thirdly, the text of the paper must be available in English. This requirement is necessary as it ensures comprehensibility and understanding of the content. Papers not available in English will be excluded from the review.

Based on the identified concepts, synonyms, and exclusion criteria, a search query has been formulated as follows:

Equation 1 Systematic literature search query

TITLE-ABS-KEY (compar OR analy* OR evaluat* OR similar* OR categori*) AND TITLE-ABS-KEY (“project planning*” OR “planning activit*”) AND TITLE-ABS-KEY (automat* OR “intelligent computing” OR AI OR “Artificial Intelligence”) AND PUBYEAR > 2016 AND PUBYEAR < 2024 AND (LIMIT-TO (LANGUAGE , “English”))*

It is essential to highlight that the base concepts for this review, shown in

Table 1, do not specifically include the AEC industry. This decision was made intentionally, as this research focuses on unsupervised machine learning and the implementation of that on project planning, which may draw insights from various fields. Therefore, exploring methods from other disciplines can be valuable. However, to ensure that essential papers related to the AEC industry are not overlooked, a validation query has been developed:

Equation 2 Validation search query

TITLE-ABS-KEY (compar OR analy* OR evaluat* OR categor*) AND TITLE-ABS-KEY (**"construction planning"**) AND TITLE-ABS-KEY (automat* OR "intelligent computing" OR ai OR "Artificial Intelligence") AND PUBYEAR > 2016 AND PUBYEAR < 2024 AND (LIMIT-TO (LANGUAGE , "English"))*

Both queries were used, and the complete results of the papers found, eliminated, and used can be seen in Figure 1.

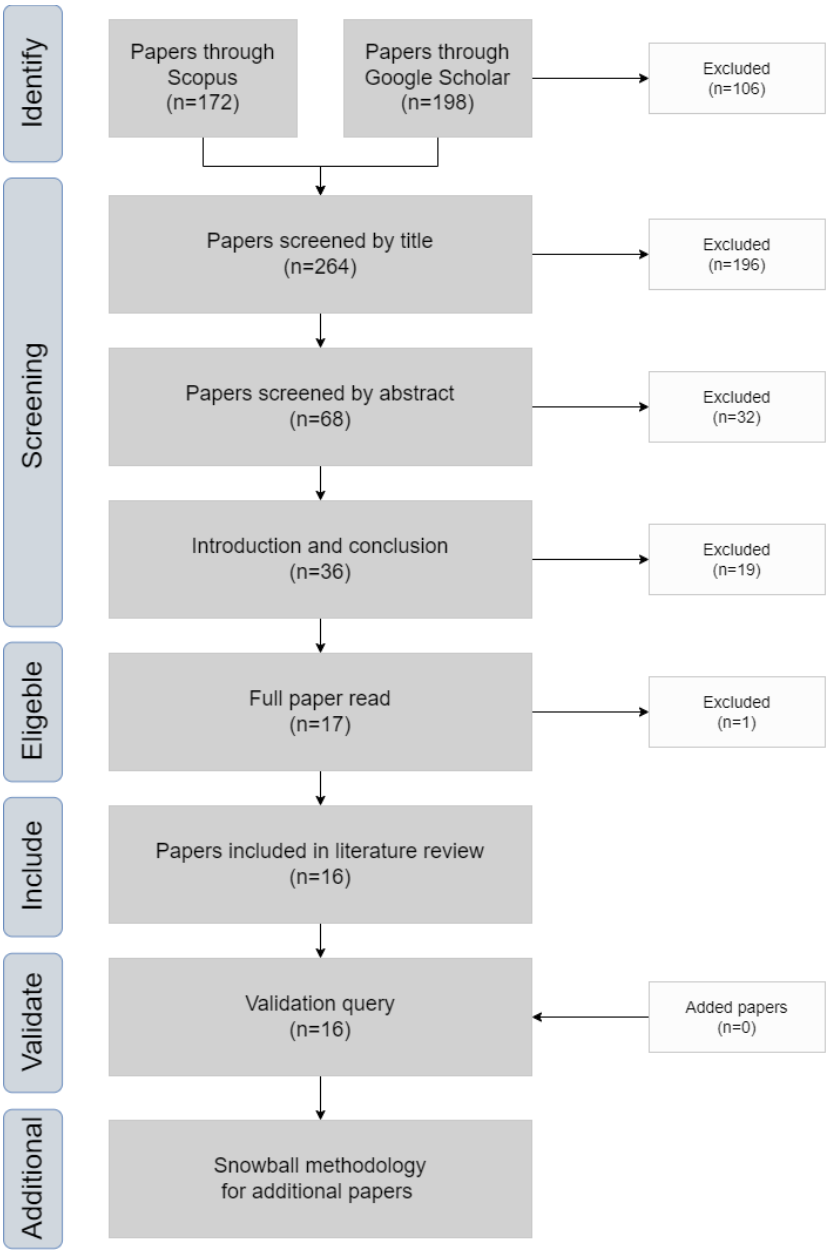


Figure 1 Paper selection process PRISMA

Table 2 shows all the literature generated after the completion of the validation. The list shows all 16 key papers read and examined on the main topics used in this literature review. It is important to note that paper 12 does not meet the inclusion criteria initially set, possibly due to an error in the exclusion process. Despite this, after examining the paper, it has been determined to include and review it in this literature analysis. However, it will not be used as a primary information source but instead used for additional information.

Table 2 List of literature with corresponding themes

	Author	Year	Project Planning	Project Control	AI	Classification Techniques
01.	Duanshun Li, Ming Lu	2019	++		+	
02.	Humyun Fuad Rahman, Ripon K. Chakrabortty, Michael J. Ryan	2020	++		+	
03.	L.S.R. Supriadi, Y. Latief, B. Susilo, M. Rajasa	2017	++		+	++
04.	Robert Pellerin, Nathalie Perrier	2019	++	++	+	
05.	Fahad Ul Hassan; Tuyen Le, A.M.ASCE; and Chau Le, A.M.ASCE	2023	++		++	
06.	Ravi Kumar Begur Nagarajappa, Yeresime Suresh	2023	++	+	++	
07.	Miklos Hajdu	2018	++	++	+	
08.	Amirsaman Mahdavian, Alireza Shojaei	2020	++	++	+	+
09.	Hossein Derakhshanfar, J. Jorge Ochoa, Konstantinos Kirytopoulos, Wolfgang Mayer, Vivian W.Y. Tam	2019	++	++		
10.	Xiao Li, Chengke Wu, Fan Xue, Zhil Yang, Jinfeng Lou, Weisheng Lu	2022	++		++	++
11.	Navid Torkanfara, Ehsan Rezazadeh Azar	2020	++		++	++
12.	Samaneh Zolfagharian, Javier Irizarry, Aziruddin Ressang, Masoud Gheisari	2014	++		++	
13.	O. Bokor, L. Florez, A. Osborne, Barry J. Gledson	2019	++	++	+	
14.	VS Pillai, KJM Matus	2020			++	++
15.	Sajjadur Rahman, Eser Kandogan	2022			++	++
16.	Shiping Wen, Handong Tang, Fei Ying, Guangdong Wu	2023	++	++	+	+

Table 2 shows the four main topics included in this literature review: Project Planning, Project Control, Classification Techniques and Artificial Intelligence. The table summarises the 16 relevant publications, their authors, publication years, and the extent to which each paper covers the identified topics. This extent of relevance is indicated by a plus sign (+). The symbol "++" signifies that a paper covers the corresponding topic. For instance, if a paper is marked with "++" under Project Planning, it indicates that it explores various aspects of project planning or goes in-depth on one aspect.

On the other hand, the symbol "+" suggests that a paper touches upon that specific topic. Papers with an empty cell indicate that the topic is not directly addressed or covered within those papers. By using

these symbols, the table allows for a quick assessment of the coverage of each topic across the selected papers.

Table 2 gives insights into the generated literature list and the corresponding topics. Project Planning emerges as a dominant theme, with a substantial number of papers (marked with "++") focusing on this topic. The reason for this is not that surprising since Equation 1 and

Table 1 both indicate “Project Planning” as one of the main concepts. Next to that, Project Control also receives less attention in the papers. This lower attention can be explained by the fact that this term is not listed as one of the main concepts. However, Project Control remains closely related to Project Planning as it involves monitoring and managing the execution of the project planning (Rozenes et al., 2006). In addition to the substantial presence of Project Planning in the papers, Artificial Intelligence emerges as a widely researched theme. Although the number of papers marked with "++" is lower for Artificial Intelligence, it is touched upon in almost all papers. The reason behind this is the same as for Project Planning, as Artificial Intelligence is likewise included in the search query and main concepts. Moreover, like Project Control, Classification Techniques are not explicitly included in the main search query or concepts. However, Classification Techniques are closely related to Artificial Intelligence and can be considered an integral part (Antonijevic et al., 2020).

2.1.1. Project planning

The general concept of planning and scheduling is not new in history. Organizing tasks and allocating resources efficiently has been an essential aspect of civilisations throughout history. During ancient times, civilizations like the Mesopotamians, Egyptians, and Romans already used planning and scheduling techniques to manage large-scale construction projects, agricultural activities, and military campaigns effectively. These early systems often relied on simple tools like calendars, hieroglyphs, and basic project management techniques (O’Brien & Plotnick, 2005). As societies evolved and became more complex, planning and scheduling techniques also advanced. In the 20th century, systematic techniques and tools in project management were introduced. Notable contributions during this time are the Program Evaluation and Review Technique (PERT) and the Critical Path Method (CPM), which helped schedule and coordinate complex projects with multiple activities (O’Brien & Plotnick, 2005). PERT, developed by the U.S. Navy in the 1950s for managing complex projects like the Polaris missile program, introduced probabilistic methods to handle uncertainty in project timelines (Roman, 1962). CPM, developed independently by DuPont and Remington Rand in the late 1950s, focused on identifying critical activities and calculating the minimum project duration (O’Brien & Plotnick, 2005).

Furthermore, the emergence of the Work Breakdown Structure (WBS) in the 1960s brought a more hierarchical approach to project planning, which helped break down large projects into smaller, manageable pieces (Zecheru & Olaru, 2016). In the 1970s, the Project Management Body of Knowledge (PMBOK) was introduced, which provided a framework for project management practices (Brotherton, 2008). The 1980s saw the introduction of project progress software such as Primavera and Microsoft Project, which facilitated scheduling, resource allocation, and project tracking. The development of the internet and digitalization of project management data in the 2000s led to the creation of tools like Relatics, which allowed for real-time collaboration and communication between project team members (Relatics, 2023).

In the context of the AEC industry, the history of planning and scheduling practices will be explored, emphasising the innovations from the 20th century onwards. Since then, construction projects have become more complex(Wu et al., 2018). Therefore, a growing need for systematic approaches to manage resources, timelines, and budgets occurred. In the early stages, planning and scheduling in the AEC industry were primarily done using manual methods such as bar charts, Gantt charts, and the CPM (Pellerin & Perrier, 2019). While these methods allowed project managers to visualize and arrange tasks, they had limitations regarding scalability and computational resources. With the advancement of computer technology in the 1970s and 1980s, construction companies began adopting computer-based tools for planning and scheduling. Software applications integrated the principles of PERT and CPM, enabling more accurate and efficient project planning. Project managers gained the ability to model project networks, estimate activity durations, and identify critical paths. As the AEC industry

continued to evolve, the Project Management Body of Knowledge (PMBOK), developed by the Project Management Institute (PMI), introduced standardized methodologies, best practices, and knowledge areas for effective planning and scheduling. Today, the AEC industry extensively utilizes sophisticated construction project management software and BIM systems. These tools allow for the efficient coordination of resources, enhanced collaboration, and improved project outcomes while incorporating the principles of PERT, CPM, and PMBOK. The industry has made significant strides in planning and scheduling practices by leveraging technology, enabling more effective project management in this dynamic field.

2.1.2. Artificial Intelligence

In the context of this thesis, which centres on the digitalization of project planning, this section delves into digitalization, focusing primarily on AI. The upcoming discussion will delve into the AI timeline, providing insights into both historical and contemporary AI models. A general introduction is provided to establish a foundational understanding of AI. The principle of AI has roots in different parts of research, e.g., mathematics, computer science, psychology, and biology (Panesar, 2021). Artificial Intelligence is the study and development of how to make computer systems perform tasks previously executed by humans(*The Oxford Dictionary of Phrase and Fable*, 2005) The Oxford Dictionary gives examples of such tasks as visual perception, speech recognition, translation, and decision-making processes. In this chapter, the timeline of AI will be first discussed. This timeline will give a clear insight into past research regarding AI and the research that is still being conducted. Next to this, Machine Learning (ML) will be discussed. ML is a subset of AI that is of importance to this research. Another subfield discussed in this chapter is Natural Language Processing (NLP).

AI timeline

The historical timeline regarding AI presented in the article by Marcus (2020), the early concept of automatons and mechanical devices provided the groundwork for later breakthroughs in AI. Foundational ideas were provided by mathematicians and logicians such as Ada Lovelace and George Boole. The birth of computing in the 1930s and 1940s, epitomised by Alan Turing's theoretical framework and the introduction of ENIAC, provided the basis for future advances in computing. The Dartmouth Conference, organised by John McCarthy in 1956, introduced the phrase "artificial intelligence" and envisioned computers that could mimic human intellect. The following decades saw the development of early AI research, with programmes such as the Logic Theorist and General Problem Solver. Symbolic AI and rule-based systems dominated the 1950s and 1960s. However, the AI Winter of the 1970s-1980s experienced a decrease in funding and interest due to failed expectations. The emphasis switched to expert systems, laying the groundwork for a revival in the 1990s. During this time, neural networks, connectionism, and statistical and machine learning methodologies emerged. The 2000s and 2010s represented the age of deep learning and big data, with developments in neural network concepts. The timeline progresses into the 2020s, with exponential development in deep learning, the emergence of specialised AI, and an increased focus on ethics and responsible AI. The paper also looks ahead to the 2020s and 2030s. Here, neurosymbolic AI and hybrid models that combine symbolic reasoning with neural networks promise to redefine cognitive capabilities. Quantum computing, expected to be available in the 2030s and beyond, can solve complex AI challenges.

Next, delving into Marcus's (2020) exploration of AI models and key factors, the paper distinguishes between classical AI and the contemporary data-driven model. Classical AI, pioneered by researchers like McCarthy and Winograd, prioritizes knowledge, internal models, and reasoning. Emphasizing common knowledge, classical AI used cognitive models to comprehend the world. Knowledge represents the importance of commonsense knowledge in building intelligent systems. Internal models facilitate a system's understanding of its environment, encompassing objects, relationships, and

properties. The emphasis on reasoning ensured the ability to draw conclusions and inferences, facilitating information processing and decision-making.

The paper also addresses the shift to the data-driven model, which is predominant in current AI approaches. Fundamental principles include the importance of data, machine learning (ML) techniques, statistical analysis, iterative processes, scalability, and automation. The emphasis on data is mentioned regarding large and diverse datasets that are crucial for training AI models. ML techniques, utilizing algorithms to learn patterns from imported data, play a pivotal role in data-driven approaches. Statistical analysis is employed to derive insights from the data, contributing to the model's learning process. Furthermore, the significance of iterative processes is mentioned, highlighting the continuous improvement and refinement of AI models through repeated cycles. Scalability is acknowledged as a crucial aspect, emphasizing the ability of AI systems to handle increasing amounts of data and computational demands. Automation is another key principle, showcasing the integration of automated processes in AI workflows to enhance efficiency and reduce manual intervention.

Building on Marcus's insights, the discussion expands to critical elements for advancing AI. The importance of solid engineering practices, encompassing redundancy, limits, and optimization algorithms sensitive to incentives, is emphasized. Addressing cultural challenges within the deep learning community, Marcus calls for a more open and collaborative multidisciplinary context to foster AI advancement. The metaphor of "seeing the whole elephant" underscores the need for a broad viewpoint, advocating for a collaborative and inclusive approach to overcome historical conflicts that have hindered AI development for decades.

The article of Panesar (2021) is used to highlight another viewpoint regarding AI. This article introduces terms like "super AI" and "self-aware AI". It classifies AI in two perspectives: Type-1 categorizes AI based on capabilities, while Type-2 categorizes it based on functionalities. In Figure 2, a short overview can be seen:

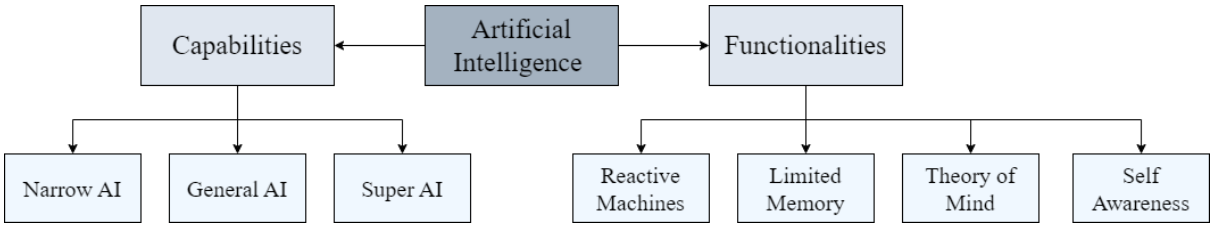


Figure 2 Types of AI

Type-1:

- Artificial Narrow Intelligence (ANI): Known as weak AI, excelling in specific tasks within a limited domain.
- Artificial General Intelligence (AGI): Aims for human-like intelligence, with the ability to understand, learn, and apply knowledge across different domains.
- Artificial Super Intelligence (ASI): The hypothetical highest level, surpassing human intelligence and potentially driving unprecedented technological progress.

Type-2:

- Reactive Machines: The basic form of AI, operating solely based on current input without memory.

- Limited Memory: Stores historical data for a short period, using it for decision-making but lacking long-term memory.
- Theory of Mind AI: Aims to understand and interact with humans by perceiving emotions and mental states.
- Self-awareness AI: The conceptual highest level, possessing subjective awareness and understanding of its existence.

Building upon the aforementioned classification of AI capabilities and functionalities, it is essential to approach discussions about "super AI" and "self-aware AI" with a critical lens. The media sometimes reacts to these ideas, adding to a sensationalised narrative about AI (Ouchchy et al., 2020). However, as Marcus (2020) points out, a nuanced and critical approach to AI is needed, which means a balanced understanding and steering away from sensationalized views.

Machine Learning

As mentioned before, ML is a subset of AI. This chapter will discuss the different types of ML (Supervised, unsupervised, semi-supervised, and Reinforced). After this part, the focus will shift to Unsupervised ML since it is used in this research. The choice for this type of ML will also be explained further. To lay the groundwork, Figure 3 illustrates the different types of machine learning, as depicted in the article of Oladipupo (2010). The different aspects of this figure are elaborated on as well.

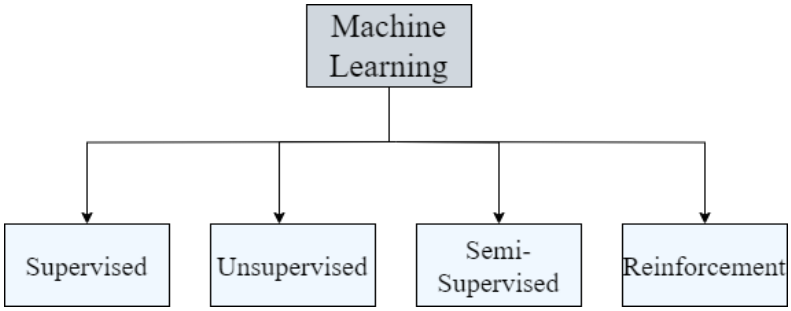


Figure 3 Types of Machine Learning (Oladipupo, 2010)

- Supervised Machine Learning:
  This approach to machine learning operates under the framework of supervision. Within this method, the database training utilizes labelled data. The program anticipates potential outputs by drawing insights from this labelled data. This approach's advantage is the reliability of the labelled training data, leading to dependable outcomes. It serves as a valuable prediction tool. Furthermore, its effectiveness relies on the quality of the imported training data. Hence, if the training data uses incorrect labels, the resulting outputs will also be incorrect. An additional drawback lies in the time-consuming nature of pre-labelling, which demands a temporal investment. Similarly, the training process itself consumes a considerable amount of computational resources. This machine learning method currently finds application in numerous domains, foremost in image segmentation, medical diagnosis, spam detection, and speech recognition.
- Unsupervised Machine Learning:
  The second approach to machine learning is known as unsupervised machine learning. As the name implies, this method operates without needing labelled or classified data. In unsupervised learning, the model trains on data without predefined categories, allowing it to independently discover patterns, similarities, and differences without human intervention. Consequently, this approach enables the system to discover its underlying patterns within the given dataset autonomously. Unsupervised machine learning encompasses two primary

categories: association and clustering. The association aims to identify relationships among data items rather than clustering them. It seeks dependencies between two data items, effectively mapping them to maximize mutual benefits. Algorithms for association rule learning include the Apriori, Eclat, and FP-growth algorithms (Mythili & Mohamed Shanavas, 2013).
On the other hand, clustering mechanisms can organize data items into categories. Various clustering methods exist, ranging from equally sized categories to hierarchical clustering. Algorithms used in unsupervised clustering include:

- K-Means Clustering algorithm
- Mean-shift algorithm
- DBSCAN Algorithm
- Principal Component Analysis
- Independent Component Analysis

The unsupervised approach is generally employed for more complex methodologies than supervised ones (Saravanan & Sujatha, 2018). The preparatory work is less time-intensive because it does not require a pre-labelled dataset. However, the outcomes may be less precise due to the absence of pre-labelled data. This is attributed to the mechanism lacking prior training knowledge. Moreover, achieving an accurate output is more challenging with this approach.

- Semi-Supervised Machine Learning:
  As the name indicates, this type of machine learning lies between the supervised and unsupervised methods. This means that this type of method combines labelled and unlabelled data. In most cases, the data consists of a few items with labels, but the gross of the items is not labelled. This method is introduced to overcome the drawbacks of the previously mentioned two methods, supervised and unsupervised. Use the predefined labels as a guideline and ensure that the whole dataset can be used even if data items are not labelled. However, a drawback of this method is its low accuracy, which may lead to results that are not stable.
- Reinforcement Learning:
  Reinforcement learning operates through a feedback-driven process where an AI agent navigates its environment via trial and error, learning from its actions to enhance performance. The agent is rewarded for favourable actions (Positive Reinforcement Learning) and/or penalized for unfavourable ones (Negative Reinforcement Learning). This machine learning type helps solve complex real-world problems that previously mentioned methods can not solve. However, the drawback is that Reinforcement learning needs large datasets and many computations. Moreover, when too much reinforcement learning is done, the results can be weakened.

Unsupervised clustering

This section will dive further into the unsupervised side of ML. Unsupervised classification is essential in this thesis since it provides valuable insight into how non-labelled data can be clustered. Therefore, the choice is made to delve further into this topic.

Below, four unsupervised clustering algorithms are described: the K-means Clustering Algorithm, Mean-shift Algorithm, DBSCAN Algorithm, and Agglomerative Hierarchical Clustering. These four algorithms can be divided into three clustering methods: hierarchical, partitioning, and density-based. Each of the algorithm methods is concisely described with the needed steps. The article by Amasha et al. (2020) describes the categories and algorithms. Based on this article, Table 3 shows the category of clustering methods to which the four described algorithms belong.

Table 3 Overview algorithms and the corresponding category (Amasha et al., 2020)

K-means Clustering	Mean shift	DBSCAN	Agglomerative Hierarchical Clustering
Partitioning	Density-based	Density-based	Hierarchical

K-means clustering

The term K-means was first introduced in an article by (MacQueen et al., 1967). The algorithm itself was researched in the 1950's by separate research efforts. All the algorithms developed in those separate research studies have four steps in common, with some other steps being different (Pérez-Ortega et al., 2020). The main goal of K-means is to divide a dataset into K districts, which are non-overlapping clusters. K-means is a centroid-based clustering algorithm. A data point will be assigned to a cluster based on the distance of this point to the centroid of the clusters. K-means, unlike DBSCAN, need the user to predetermine the K number of clusters.

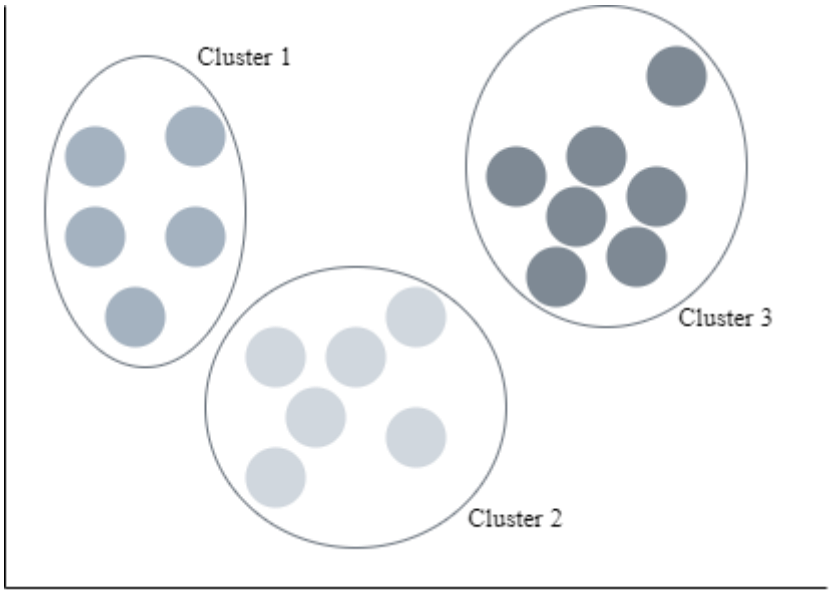


Figure 4 K-means clustering visualization

The K-means algorithm begins by randomly distributing K cluster centroids throughout the data. These centroids will serve as the representative cluster centres of each K cluster. An important thing regarding this algorithm is that the K, the number of clusters, must be determined beforehand. Subsequently, each data point in the dataset is assigned to the nearest cluster centroid. This is done using a chosen distance metric. Most often, this will be the Euclidean distance. The algorithm subsequently adjusts

the centroids for each cluster by determining the mean (average) position of all the data points assigned to that particular cluster. This step is essential to ensure that the centroids accurately represent the characteristics of the data within their respective clusters. The process is then repeated iteratively, with data points moved to the closest centroid following the new centroid positions. This process stops when convergence is reached, which happens when there is little to no change in the assignments of the data points or when a predetermined maximum number of iterations is reached, this can be seen in Figure 4. These assignments and centroid updates repeat based on the selected number of clusters (K). The data is clustered most optimally.

Mean Shift

Mean shift is a clustering algorithm used in real-world data analysis, such as segmentation (Jordan & Angelopoulou, 2013). This algorithm was first introduced by Fukunaga & Hostetler (1975) and developed over the years.

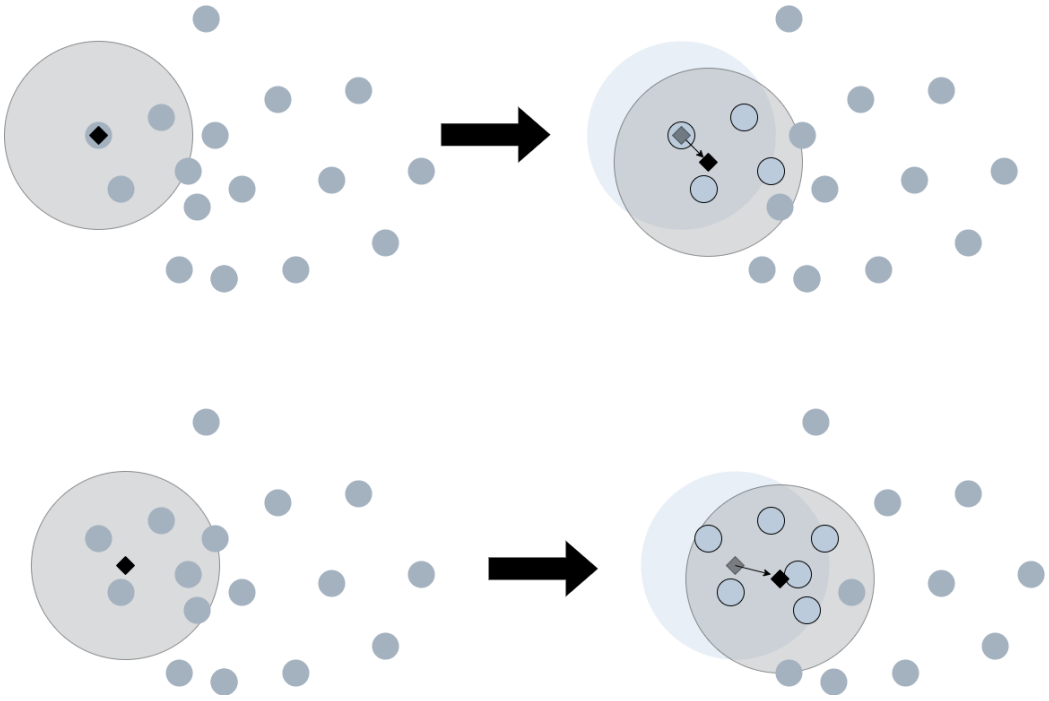


Figure 5 Mean-shift visualization (Sippel et al., 2015)

Unlike K-means, Mean Shift does not require specifying the number of clusters beforehand, making it a more flexible algorithm for discovering the natural structure of the data. Mean Shift operates by iteratively shifting data points towards the mode or peak of the underlying probability density function, effectively converging towards the cluster centres (Sippel et al., 2015). The mode is determined by the local density of data points, making it adaptive to the shape and density of the data distribution. The algorithm starts by placing a kernel (a window function) at each data point. For each iteration, it computes the mean shift vector, pointing towards the direction of the highest increase in the local density within the kernel. The data points are then updated by shifting them along this mean shift vector. Figure 5 illustrates the Mean Shift clustering process. As the algorithm progresses, data points converge towards the modes of the data distribution, forming natural clusters without the need for a predetermined number of clusters.

Mean Shift's ability to adapt to the data distribution and automatically discover the number of clusters makes it particularly useful in scenarios where the underlying structure of the data is not well-known beforehand. However, its computational complexity can be higher, especially for large data sets.

DBSCAN

DBSCAN is a density-based clustering algorithm. In an article from (Ester et al., 1996), the concept of DBSCAN, a density-based clustering algorithm, was first proposed. This article mentions that the algorithm operates assuming that clusters are separated by lower-density regions in space. One of the unique points of the DBSCAN algorithm is that it is robust to outliers (Mahesh Kumar & Rama Mohan Reddy, 2016). Furthermore, the algorithm does not need a set number of clusters beforehand. However, it does need two parameters beforehand: Epsilon and minPoints. As Figure 6 shows, DBSCAN forms a circle for clustering, with epsilon being the circle's radius and minPoints the minimum number of data points inside the circle to form a cluster.

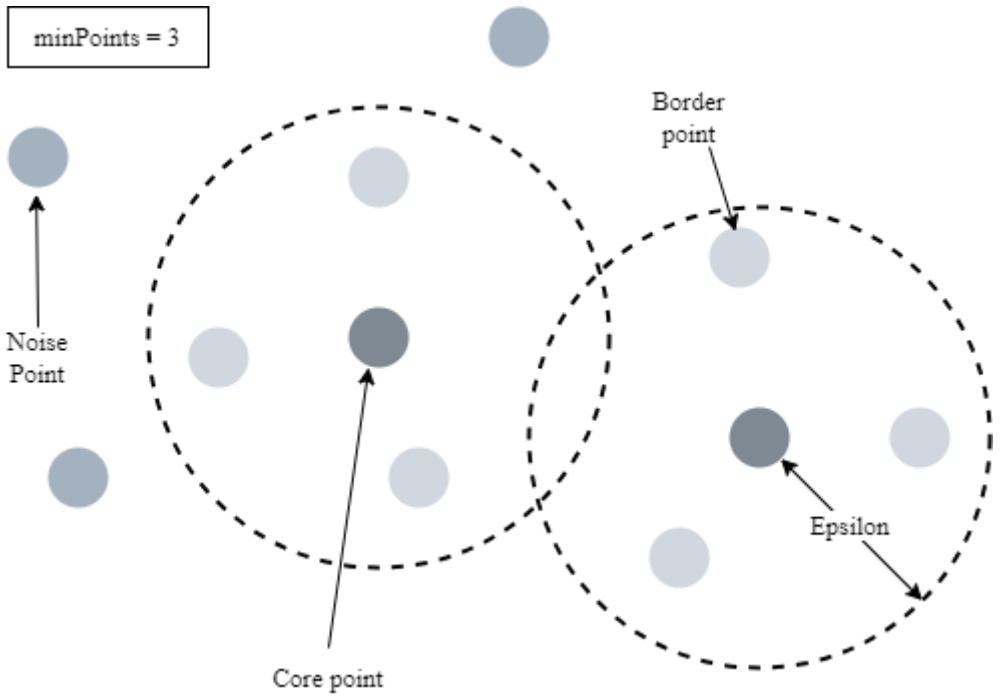


Figure 6 DBSCAN parameters visualisation (Pandey, 2020)

Based on these two parameters that need to be defined beforehand, the data points are classified as one of three types of points. The first is the core points, which can become a core point if the minPoints parameter is met in the Epsilon (radius) from this point. The second type of point is the border point. A border point is a data point in a core point's cluster. A border point has fewer than minPoints within its Epsilon radius. The last type of point is a noise point. A noise point is a point that does not fall in the cluster and does not meet the criteria of a core point. All these points can also be seen in Figure 6.

Agglomerative Hierarchical Clustering

The key idea of this method is to start with individual data points as separate clusters and iteratively merge them until a single cluster representing the entire dataset, is formed (Bridges, 1966).

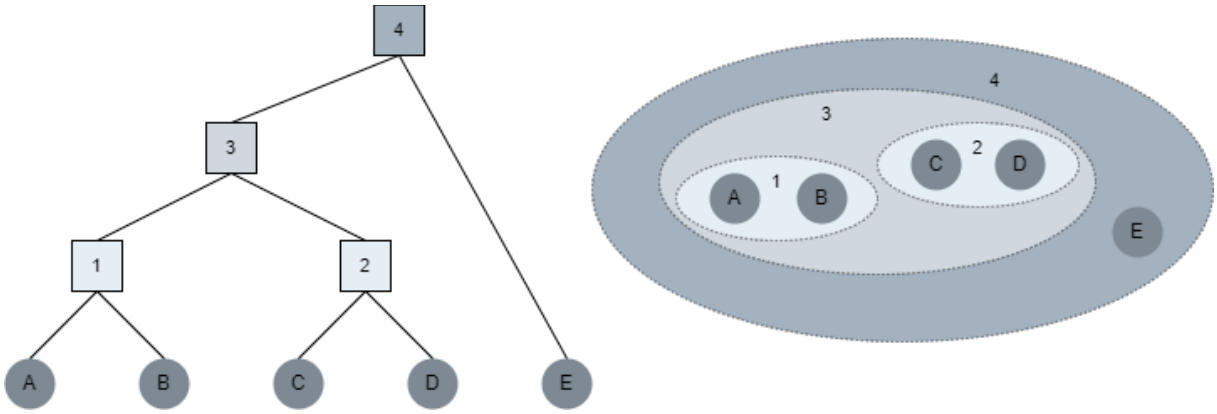


Figure 7 Hierarchical clustering visualization (Nielsen, 2016)

The process begins by treating each data point as a singleton cluster. The algorithm then iteratively merges the closest clusters based on a chosen linkage criterion, such as Ward's method, complete linkage, or average linkage (Murtagh & Legendre, 2014). The linkage criterion determines the distance between clusters and guides the merging process. The distance between clusters is calculated based on a chosen metric, commonly Euclidean distance. As clusters are merged, the hierarchy is represented by a dendrogram, a tree-like structure that visually depicts the merging sequence and distances. The user can determine the number of clusters by setting a threshold on the linkage distance or cutting the dendrogram at a specific height. This flexibility allows for the exploration of different cluster granularity levels. Figure 7 illustrates the Agglomerative Hierarchical Clustering process. Initially, each data point forms its own cluster, and through successive iterations, clusters are merged until a single cluster encompasses the entire dataset.

Agglomerative Hierarchical Clustering is advantageous for capturing hierarchical relationships within the data, providing insights into fine-grained and coarse-grained structures. However, its computational complexity should be considered, particularly for large datasets.

Natural Language Processing

As mentioned at the beginning of this chapter, NLP is also a subset of AI. Since this research will delve into the automated evaluation of project planning, this subset of AI is of great importance. In general, a project consists of text. The above-mentioned machine learning algorithms have one main thing in common: they are based on mathematical principles (Lamba et al., 2021). They draw from concepts like linear algebra, matrix manipulation, and calculus. However, applying these mathematical rules becomes considerably challenging when confronted with textual data as input. This challenge is particularly relevant in the field of Natural Language Processing (NLP), where the primary data source is text.

In the NLP book of O'Connor (2001), it is described that NLP consists out of six basic principles. First and foremost is recognising that individuals play a central role in using NLP. Within the book, this principle is referred to as "You". "You" underscores the significance of one's emotional state and skill level. The second principle includes the presuppositions of NLP. This focusses on foundational beliefs that underpin the NLP practices. Rapport, the third principle, emphasises the importance of quality relationships in establishing mutual trust and responsiveness. Understanding and respecting the perspectives of others is essential for effective communication. The fourth principle is "Outcome," which highlights goal clarity and the ability to communicate the desires of others. Understanding the current situation, imagining the desired situation, and planning the strategy to bridge the gap are all part of "Outcome". The fifth principle, Feedback, underscores the importance of paying attention to sensory input to evaluate progress and make informed decisions. Finally, the sixth principle, Flexibility, highlights the need to adapt strategies if initial approaches prove ineffective. NLP encourages success through purposeful action within the context of rapport and awareness by promoting diverse emotional states, communication styles, and perspectives. These principles form the necessary framework for effectively implementing NLP techniques and interventions.

The process of text vectorisation becomes essential to bridge this gap between textual and numerical data and enable the use of traditional machine learning algorithms with textual input. Text vectorization refers to various techniques to convert textual data into numerical representations, thereby facilitating the application of mathematical models and algorithms in NLP tasks.

- One-Hot Encoding (OHE): This way of vectorization uses tokens of 1 or 0. Using this method, each word in a set vocabulary gets a unique token of 1 (Seger, 2018). The other positions in the vector get set to 0. A disadvantage of this method is that it results in high-dimensional vectors in datasets with extensive vocabularies, which can lead to computational inefficiencies (Rodríguez et al., 2018). Furthermore, it fails to capture the nuanced relationships and contextual information between words, making it less suitable for advanced NLP tasks that demand a deeper understanding of language semantics.
- Count Vectorizer: This method simplifies the representation of textual data and generates a Document-Term Matrix (DTM), a set of binary variables indicating the presence or absence of specific words in a document. The Count Vectorizer creates a vocabulary of words to form a matrix, where each cell represents how often a word appears in a document. In which a document is a piece of text (e.g. sentence, paragraph). The dimensions of matrix M are defined as D × N, where D signifies the number of documents, and N represents the count of unique tokens extracted from the corpus (Kulkarni & Shivananda, 2021). Each row in the matrix corresponds to a document, while columns are dedicated to individual words found in the vocabulary. This method ensures the flexibility to handle extensive vocabulary by focusing on the essential words in the data.

- Bag-of-Words (BoW) converts textual content into numerical feature vectors. Its process involves taking a document from a corpus and transforming it into a numeric vector (Kulkarni & Shivananda, 2021). Each word within the document is mapped to a feature vector for use in machine learning models. Tokenization and vector generation are the two main steps in this text vectorization technique (Perkins, 2014). Tokenization requires deconstructing phrases into words or other smaller tokens. The complete corpus, which contains the tokens from every document and serves as the basis for building the bag of words, is mined for all the unique words after tokenization. The vectorization step involves generating vectors for every document. The size of the vector for a specific document equals the count of unique words found in the corpus. For each document, the vector entries are populated with the corresponding word frequencies for that particular document. It is important to note that the Bag of Words method has also disadvantages. It does not preserve word order, making it unsuitable for tasks reliant on word sequence information. Additionally, it does not enable the derivation of meaningful insights for downstream NLP tasks.
- N-grams Vectorization: This involves the generation of a document-term matrix where each cell holds a count of occurrences. In this technique, the columns represent sequences of adjacent words of a specified length 'n'. Count vectorization can be seen as a specific instance of N-Gram when 'n' is set to 1 (Kulkarni & Shivananda, 2021). Unlike the BoW approach, N-Grams consider the order of words in the text, allowing for a more nuanced analysis. The choice of the 'n' value depends on performance considerations, creating a trade-off. Selecting a smaller 'n' might not yield sufficient information, while a larger 'n' leads to a substantial matrix with many features. Hence, N-Grams offer powerful capabilities but require careful parameter tuning. Nevertheless, N-Grams have certain drawbacks, including generating numerous features, resulting in computational complexity, and the challenge of determining the optimal 'n' value.
- TF-IDF Vectorization: This method transforms textual data into a numerical format by generating a document-term matrix. In this matrix, each cell represents a numerical value calculated using the TF-IDF formula (Kulkarni & Shivananda, 2021). Unlike count vectorization or N-Grams, TF-IDF considers both term frequency (TF), which measures how often a term appears in a document, and inverse document frequency (IDF), which evaluates the importance of a term within a corpus of documents. In TF-IDF vectorization, the columns correspond to unique terms or words in the entire corpus, while the rows represent individual documents. The numerical values in the cells reflect the TF-IDF scores, signifying the relevance of a term within a specific document concerning its occurrence in other documents in the corpus. High TF-IDF scores indicate that a term is both frequent within the document and unique to that document, thus emphasizing its significance. The TF-IDF vectorization approach offers the advantage of capturing the importance of terms while considering their frequency in the context of the entire corpus. This helps distinguish between common words and terms that carry specific meanings within individual documents. However, it also introduces considerations such as parameter tuning, including the choice of TF-IDF variants and strategies for handling rare terms, which can impact the overall performance of the vectorization process.

2.1.3. AI based planning

In this systematic literature review, the aspects of planning and AI have been addressed separately. This section will address the common ground between AI and planning. In recent years, the field of automated planning has gained more attention, resulting in an increased amount of studies on this topic. This chapter uses the papers on the PRISMA method to analyse the different approaches regarding AI and digitalization in project planning. Table 4 shows these papers.

Before delving into the nuanced examination of the chosen papers, it is critical to explain the underlying rationale for establishing a link between artificial intelligence (AI) and the planning field within this research's scope. The research objectives state that the primary goal is to improve planning evaluation through historical data. This research goal is directly connected to the fundamental principles of AI, as discussed in the previous section. Recognising this natural connection, a decision was made to enter this area of digitalization. To achieve the goal stated in the objectives, the study aims to use AI's computational powers to systematically process historical data, allowing for detailed and reliable assessments of planning techniques.

In order to achieve this, the digitalization of AI tools discussed in other papers is analysed. All the papers are divided into four different categories. Below are these clusters and their main findings.

Cluster 1: Automatic Planning
Several writers have explored the subject of automated planning to speed up the detection and solution of how to handle conflicting activities. With the ultimate goal to generate a way to make automatic planning work. In a study by Li & Lu (2019), automatic identification of clashing activities took centre stage. Likewise, Pellerin & Perrier (2019) showcased the utility of Natural Language Processing (NLP) in scheduling tasks, aligning with the automated planning strand. This cluster highlights the use of advanced algorithms and AI techniques to automate planning processes. However, the drawback of these papers is that automatic planning only works in specific fields and areas, with some assumptions and side notes, which means that automatic planning that can be used on a larger scale is hard to accomplish.

Cluster 2: Constraint-Based Planning
Another cluster emerges from papers focusing on constraint-based planning methodologies. (Hassan et al., 2023). They introduced the concept of automated digital scope of work, while Mahdavian & Shojaei (2020) emphasized the role of Discrete-Event Simulation (DES) in shaping scheduling frameworks. Collectively, these studies explore the integration of constraints and simulations to enhance planning accuracy and efficiency.

Cluster 3: Standardized Work Breakdown Structures (WBS)
This cluster is centred around standardized Work Breakdown Structures (WBS). It reflects efforts to establish consistent frameworks for project planning. Sagita et al. (2017) delve into standardized WBS, while Nagarajappa & Suresh (2023) examine the pursuit of consistency in Activity Definition within the WBS. Torkanfar & Rezazadeh Azar (2020) delve into similarity assessments of WBS, collectively contributing to the innovation on standardized planning structures. These papers already take a step back from the automated planning approach; however, they focus more on the WBS than the project planning itself.

Cluster 4: Classification and Machine Learning
The importance of classification and machine learning methodologies emerges from the work of Hajdu (2018), who conducted a classification-based survey, and Derakhshanfar et al. 2019), who employed

machine learning for risk overview. Wen et al. (2023) and Rahman & Kandogan (2022) further champion the role of AI-driven classification and information extraction in refining planning processes.

Table 4 Planning innovations within read papers

	Author	Year	Innovation	Type
01.	Duanshun Li, Ming Lu	2019	Automatic identification of clashing activities	Automatic planning
02.	Humyun Fuad Rahman, Ripon K. Chakrabortty, Michael J. Ryan	2020	RCPSP generates a schedule	Constraint-based planning
03.	L.S.R. Supriadi, Y. Latief, B. Susilo, M. Rajasa	2017	Standardized WBS	Standardized WBS
04.	Robert Pellerin, Nathalie Perrier	2019	NLP used in scheduling	Automatic planning
05.	Fahad Ul Hassan; Tuyen Le, A.M.ASCE; and Chau Le, A.M.ASCE	2023	Automated digital scope of work	Constraint-based planning
06.	Ravi Kumar Begur Nagarajappa, Yeresime Suresh	2023	Consistency in Activity Definition in WBS	Standardized WBS
07.	Miklos Hajdu	2018	Classification based on survey	Classification
08.	Amirsaman Mahdavian, Alireza Shojaei	2020	Discrete-event simulation (DES)	Constraint-based planning
09.	Hossein Derakhshanfar, J. Jorge Ochoa, Konstantinos Kirytopoulos, Wolfgang M ayer, Vivian W.Y. Tam	2019	Machine learning used for risk overview	Classification
10.	Xiao Li, Chengke Wu, Fan Xue, Zhil Yang, Jinfeng Lou , Weisheng Lu	2022	Grouping of similarities based on questions	Classification
11.	Navid Torkanfara, Ehsan Rezazadeh Azar	2020	Similarity assessment of WBS	Standardized WBS
12.	Samaneh Zolfagharian, Javier Irizarry, Aziruddin Ressang, Masoud Gheisari	2014	Automatic planning with AI	Automatic planning
13.	O. Bokor, L. Florez, A. Osborne, Barry J. Gledson	2019	automatic work packaging in modular construction projects	Automatic planning
14.	VS Pillai, KJM Matus	2020	WBS categorization	Standardized WBS
15.	Sajjadur Rahman, Eser Kandogan	2020	Information extraction with AI	Classification
16.	Shiping Wen, Handong Tang, Fei Ying, Guangdong Wu	2023	Classification with AI	Classification

Exploring various research papers on planning methods, a significant insight can be seen: the attention around automated planning is accompanied by a web of complexities. The assumptions and dependencies inherent in these techniques raise a valid concern: a single approach might not

seamlessly fit the complexity of real-world projects. However, through the conduction of this literature research, it can be seen that innovation goes beyond both manual planning and current automation. The AEC industry, often characterized by a lack of digitalization, presents a promising industry to start digitalizing. This journey entails a shift that is highlighted using the previously mentioned clusters. The papers in those clusters focus on data-driven analysis, classification techniques, constraint-based strategies, and their overlap. These different topics have already sparked the idea of more digitalization within the AEC sector. It is already a step in the direction of automated planning. However, the sector is not yet there. A common finding in all these papers is that automated planning is still a few steps ahead. As Table 4 marks, the shift needed to come to the automated planning outcome first needs a good and working semi-automated planning method. One that will help planners but does not all the work for them.

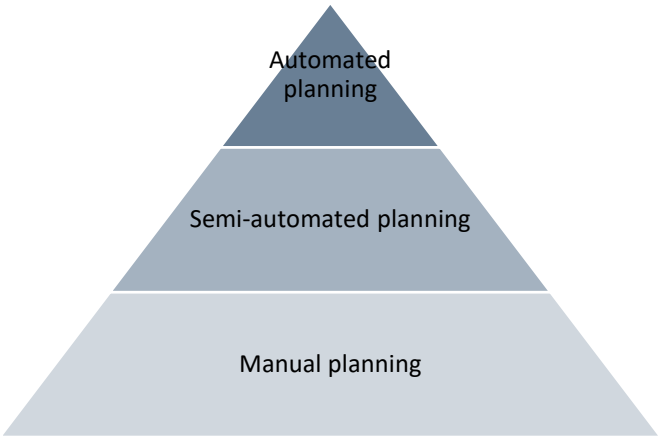


Figure 8 Automation in planning pyramid

2.2. Bibliometric literature review

The second step in this literature research is to conduct a bibliometric analysis. This quantitative analysis will help to see the relationships between the different articles from the used search query. Generally, a bibliometric analysis refers to a quantitative approach that evaluates and analyses the characteristics and patterns of the scientific papers used. This type of analysis includes the systematic collection, measurement, and statistical analysis of bibliographic data, such as citation counts, publication trends, authorship patterns, and journal impact factors (Bouyssou & Marchant, 2011; Yang & Lee, 2013). Examining these bibliometric indicators makes it possible to gain insight into aspects of the literature, like identifying influential authors, main topics within papers, or papers with a high citation rate. Bibliometric analysis can serve as a valuable tool for identifying research gaps, evaluating scientific productivity, and recognising contributors within the researched field (Deng et al., 2009; José de Oliveira et al., 2019). Furthermore, it can identify research gaps or validate qualitative literature reviews (José de Oliveira et al., 2019).

In this particular instance, the bibliometric analysis incorporates the examination of keyword co-occurrence as a fundamental component. Keyword co-occurrence analysis involves assessing the frequency, patterns, and relations of keywords of different papers (Surjandari et al., 2015). This approach will uncover interrelationships, thematic clusters, and knowledge networks embedded within the selected papers by identifying and analysing the co-occurrence of specific keywords.

The bibliographic literature review focuses on analysing keyword co-occurrence patterns. The search query presented in section 2.1.1 Methodology of the systematic literature review is used to generate

a bibliometric keyword co-occurrence map. As previously mentioned in section 2.1, the literature consists of four main themes: Project Planning, Project Control, Artificial Intelligence, and Classification Techniques. Project Planning and Artificial Intelligence emerge as the primary focal points among these themes. This research explores the quantitative aspects of these two themes by examining articles that predominantly revolve around them.

Consequently, two distinct bibliometric maps will be generated—one encompassing all the articles from the search query in Equation 1 that include the keyword Project Planning, and another focusing on articles containing the keyword Artificial Intelligence. This chapter includes the maps made with VOSViewer and a concise conclusion of the maps. Appendix A can be consulted for the full explanation of the maps and clusters made within VOSViewer.

2.2.1. Project Planning

The bibliometric map, shown in Figure 9, is based on the articles focusing on Project Planning. Within this map, six distinct clusters can be found. In general, the most common words are project planning, artificial intelligence, and project management.

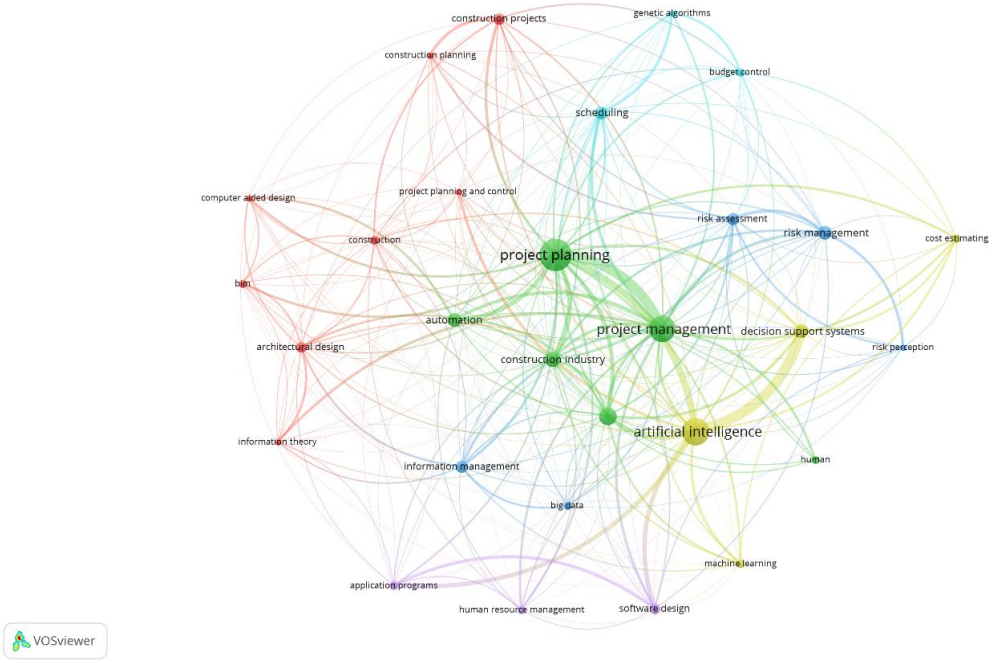


Figure 9 Bibliometric map of Keywords of Project Planning articles based on occurrences

The analysis covers several clusters, each emphasizing different focal points within the reviewed literature:

Cluster 1 (Red): Emphasizes architectural design, construction projects, project planning, and control. Technological advancements like BIM and CAD are highlighted, focusing on efficient and effective construction planning.

Cluster 2 (Green): Concentrates on project planning, management, decision-making, and automation within the construction industry. It strongly emphasises planning processes, effective project management, and integrating automation to improve efficiency.

Cluster 3 (Blue): Focuses on risk management, assessment, and perception. It delves into understanding and mitigating risks, utilizing information management and data analytics, mainly through the 'Big Data' lens.

Cluster 4 (Yellow): Centres around Artificial Intelligence (AI) applications, especially in decision support systems, machine learning for cost estimation, and its impact on construction projects.

Cluster 5 (Purple): Explores software design, application programs, and human resource management in project planning, indicating research not solely related to the Architecture, Engineering, and Construction (AEC) industry but also industries intersecting with project planning and AI.

Cluster 6 (Aqua): Highlights scheduling, budget control, and the application of genetic algorithms within project management, illustrating the focus on optimizing project timelines and resource allocation.

The analysis also outlines significant associations between key concepts:

- **AI and Decision Support Systems:** This signifies a strong association, suggesting the use of AI in developing decision support systems.
- **Project Planning and Project Management:** Demonstrates a close relationship, highlighting the critical role of effective planning in successful project management outcomes.
- **AI and Project Planning:** Indicates a connection between AI techniques and their application in project planning processes, extending AI's influence beyond decision support systems.

The strong link strengths reported across these clusters suggest significant connections within the examined literature. The findings indicate potential prospects for using advanced technology and computational techniques to improve project decision-making, resource allocation, and risk reduction.

2.2.2. AI

The bibliometric map, shown in Figure 10, is based on articles focusing on Artificial Intelligence. Within this map, five distinct clusters can be found. In general, the words that have the most occurrences are Artificial Intelligence, Project Planning, and Decision Support Systems.

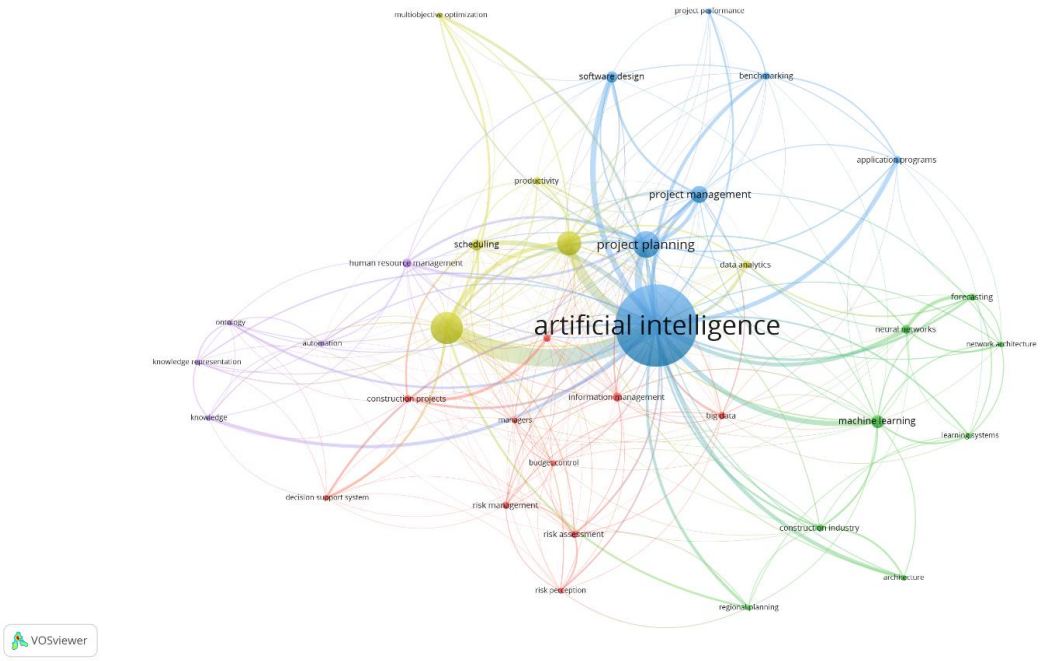


Figure 10 Bibliometric map of Keywords of AI articles based on occurrences

Cluster 1 (Red): Emphasizes information management, big data, risk assessment, risk management, decision support, and the role of project managers within construction projects. It suggests exploring managing information effectively and mitigating risks using data-driven decision support systems.

Cluster 2 (Green): Concentrates on machine learning applications within the Architecture, Engineering, and Construction (AEC) industry. It covers machine learning algorithms, neural networks, forecasting, learning systems, and network architecture, focusing on their integration into construction processes.

Cluster 3 (Blue): Highlights the prevalence of Artificial Intelligence (AI) as the main topic. It showcases research on AI techniques for project planning, project management, software design specific to construction, application programs, benchmarking, and project performance evaluation.

Cluster 4 (Yellow): Demonstrates a strong emphasis on decision support systems, decision-making, scheduling, data analytics, productivity enhancement, and multi-objective optimization within construction projects. This cluster indicates research focusing on informed decision-making and resource optimization.

Cluster 5 (Purple): Focuses on human resource management, automation, knowledge utilization, and representation within construction projects. It emphasizes the management and impact of human resources, the adoption of automation, and knowledge utilization and representation strategies.

The observed link strengths between key concepts in this analysis:

- **AI and Decision Support Systems:** Indicates a significant association, emphasizing the interdependence between these two areas in enhancing decision-making processes.
- **AI and Decision Making:** Strengthens the connection between AI and decision-making processes, emphasizing the integral role of AI in facilitating informed decision-making.
- **AI and Project Planning:** Demonstrates a connection between AI techniques and their application in project planning, extending AI's influence beyond decision support systems into the planning phase of projects.

These findings suggest a considerable amount of associations and connectivity within the analysed literature, with an integrated research landscape encompassing themes like AI, decision support systems, decision-making, machine learning, and their interplay within various aspects of construction projects.

2.3. Summary of literature review results

This chapter provides an overview of the literature on project planning, AI, and AI-based planning. Setting the stage for the subsequent chapter on research methods. The combined findings of the two studies, shown in this chapter, reflect an interrelated research environment in construction projects and project management. They emphasise the importance of using technology, information management, and data-driven decision-making to improve project results, minimise risks, and optimise different construction industry processes. The interaction of artificial intelligence, decision support systems, machine learning, and project planning highlights the potential for developments in resource allocation, decision-making, and overall project efficiency and effectiveness. Both analyses provide useful insights into the varied techniques and problems in the current AEC industry. This section begins by presenting a list of 16 key research papers and their corresponding themes, which include Project Planning, Project Control, Artificial Intelligence, and Classification Techniques.

The chapter highlights Project Planning and AI as central themes, with numerous papers focusing on this area. It also highlights the connection between Project Control and Project Planning. The emergence of Artificial Intelligence as a widely researched topic, coupled with the relationship between Classification Techniques and AI. This represents the growing importance of digitalization and innovation in the AEC sector. The historical development of planning and scheduling practices in the AEC industry is traced, highlighting the introduction of systematic techniques like PERT, CPM, and the Work Breakdown Structure (WBS). The impact of the Project Management Body of Knowledge (PMBOK) and the digitalization of project management data is also discussed. The chapter delves into the field of Artificial Intelligence, covering the AI timeline and its subsets, ML and NLP.

The main findings of the literature review are categorized into four clusters: automatic planning, constraint-based planning, standardized work breakdown structures (WBS), classification, and machine learning. These clusters offer insights into the complexities and challenges of automated planning in the AEC industry.

In conclusion, this chapter highlights the need for digitalization and innovation in the AEC sector to advance towards automated planning processes. It suggests that while there is promise in various research directions, the industry may need to establish effective semi-automated planning methods as a stepping stone toward full automation. The next chapter will discover these principles, where unsupervised ML will be elaborated on.

3. Methodology

This chapter will dive into the steps needed to answer the research questions on how AI can be used in the evaluation of the activities of large-scale planning projects. The literature review highlighted the timeline regarding both project planning and AI. As well as the implementation of AI in project planning. It was identified that AI in project planning research mostly focuses on automatically generating project planning. However, the issue discovered with this is that several assumptions need to be made, which means that the outcome of these research papers mostly applies to a specific domain. This has resulted in the scalability to other projects becomes hard. Therefore, this research will take a step back to see if the evaluation between activities in the planning of different projects is possible. Data can be gathered from historical planning, and an algorithm can be trained with these historical data. To compare different project plans to each other, a common ground needs to be created. In general, the activities within projects are not identical. However, most of the time, there is overlap in activities within projects. This overlap can form the basis for evaluating planning activities across different projects. A valuable insight that could be generated is to see if activities are planned realistically since the planning activities of one project can be compared to the same activities of another project that already has been completed. The ultimate goal is to generate a large database in which a new planning can be loaded. This database will tell if activities are planned within an acceptable range. This thesis will provide the first steps in generating this database. This will be done by experimenting with a method to generate these clusters, which will serve as the basis for the evaluation. Furthermore, this research will also look into how a new planning can be evaluated by assessing the right cluster for each activity. To achieve this, the methodology will explain the steps taken. Every step will be individually elaborated in this chapter.

3.1. Unsupervised ML algorithms

As the literature review explains, unsupervised ML can be a valuable tool to cluster text. Therefore, the decision is made to dive into this aspect of cluster planning activities. The literature review highlights four main clustering algorithms (Fuchs & Höpken, 2022; Jeena et al., 2023; Ren et al., 2014). Within this research, two algorithms will chosen on which a sample dataset will be tested. A Multi-Criteria Analysis (MCA) is conducted to choose the algorithms that will be tested.

Different Criteria are needed to score the different options to perform this MCA. The criteria are developed to ensure the algorithm is suitable to achieve the objectives of this research. The criteria, together with their explanation, are shown below:

1. **Quality of Clustering**: How well do the clustering techniques group similar activities together in meaningful clusters? This criterion assesses the accuracy and cohesion of the resulting clusters. This is needed to ensure that the same activities are placed in the cluster, as it is one of the main aspects of this research.
2. **Robustness**: How stable are the clustering results when the algorithm is run multiple times with different random initializations? A robust algorithm should produce consistent results. This is needed to guarantee coherence when using the algorithm for multiple projects.
3. **Scalability**: How well does the algorithm handle large datasets? Scalability is crucial since the algorithms will first be tested on a sample dataset and later used on a larger dataset. Furthermore, it is also important that the algorithm is scalable to be used in other projects as well.
4. **Interpretability**: How easy is it to interpret the resulting clusters? An interpretable clustering is important for understanding and using the clusters effectively. This is important for the outcome of this research. However, it is also important if this algorithm is used later on for other projects.

5. **Ability to Handle Varying Cluster Shapes and Sizes**: Since the clusters can be generic and have varying shapes and sizes, this criterion evaluates how well the algorithm adapts to different cluster structures. Within a project, not every activity is present in the same amount. One activity can be in planning multiple times, while the other can for example, only be present two times. The algorithm should, therefore, be able to handle different cluster sizes.
6. **Computation Time**: How long does the algorithm execute on the given dataset take? This is important to know since the dataset can be quite extensive. Waiting long on results is preferably not wanted.

Based on these criteria, the four suggested clustering algorithms can be evaluated. The rating used in this multi-criteria analysis is 1 to 5, with 5 being the highest and 1 being the lowest score, to assess each technique's performance. The scores are given based on the sources provided in the literature review. Furthermore, for every algorithm, the total score is given. This final score will decide which of the algorithms will be tested. Explanation of tested algorithms

In order to test this data sample, the algorithms must be made. The result of this code will be shown in Chapter 4 Results. This chapter explains the method of the general steps within the code. To explain which steps are needed, Figure 11 is made. The figure indicates that the first three steps for the different algorithms are identical. These steps are there to get the data into the right format for the different clustering algorithms. In the next section, the steps are individually explained.

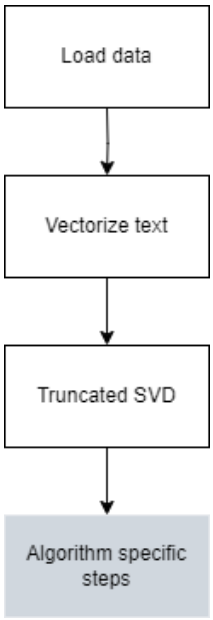


Figure 11 General steps in method of algorithms

The method starts with loading the data into Python. As the literature chapter explains, the data will be converted from textual to numerical data. The literature chapter already elaborated on different methods to do this. Different methods are tested to choose the correct way of vectorization. The BoW, N-grams, and OHE methods are not tested. For the BoW, the reason is that this method does not preserve the word order. In this research, planning activities will be vectorized. Therefore, the words belonging to a singular activity must stay together. The method does preserve the word order for N-grams, but it represents the word sequence by an identified " n " length. Since the data will consist of different lengths of words and activities, generating numerous features with N-grams could lead to computational inefficiencies and potentially require extensive parameter tuning to find the optimal 'n'

value. For OHE, the problem comes with the scalability of the algorithm. OHE is a binary vectorization technique where each word in a set vocabulary is assigned a unique binary token (1 or 0).

While it is a straightforward method, it has limitations for large datasets with extensive vocabularies. Therefore, it is also not suitable for this research. Therefore, the count vectorizer and TF-IDF methods will be tested, and the choice of vectorizer will be explained in the Results chapter. The last step of the identical steps for both methods is using the Truncated SVD technique. This step produces a new set of 2-dimensional vectors(Michałowska-Kaczmarczyk & Michałowski, 2014). Reducing vectors is important to have less computational time, use less storage, and reduce noise. These compressed vectors are used afterwards to create a Matplotlib scatter plot with the first and second components as the x and y axes, respectively. The plot is finally presented for visual representation.

3.2. Data preparation

After deciding which algorithm will be used, the complete raw data will be imported into this algorithm. This will give a clear understanding of what the starting point of the clustering is. Ultimately, it is important to get as correctly clustered outcome as possible. In this thesis, this will be referred to as cluster similarity. To improve cluster similarity, multiple options can be taken.

1. Choice of algorithm
2. Choice of dimensions
3. Data preparation

The first two options have been addressed previously in this chapter. This thesis will focus on improving the cluster similarity by adjusting the data put into the algorithm. This process will be called data preparation for the remainder of this thesis. The data preparation is part of the previously explained NLP field. This section will focus on the processes taken to go from raw data to the most optimal data preparation. In this research, the original data consists solely of the text provided by a planner for an activity. It is important to note that this textual input can vary significantly, with some containing rich contextual information in the name, while others may provide very little insight. Therefore, a uniform way of loading the data into the algorithm must be created. This chapter will focus on data preparation to optimize data clustering eventually. The different data cleaning options used are:

1. Identification of parts of speech and removing parts (e.g. adjectives, pronouns, conjunction)
2. Remove punctuation marks and text in between brackets
3. Transform words back to their stem or lemmatize them (e.g. discovered' to 'discov')
4. Language detect activities in Dutch
5. Remove project-specific words and abbreviations

The data preparation plays a crucial role in achieving optimal data clustering, a central component of this analysis. This process holds several key benefits. Firstly, the data preparation enhances the clustering accuracy by eliminating noise, irrelevant words, and language structure variations (Hintze, 2017). It allows the clustering algorithm to focus on the essential characteristics of the data, resulting in more accurate and meaningful clusters (Jeatrakul et al., 2010). Secondly, more uniformity is created by preparing the text input across all activities in the data. This consistency is needed to identify patterns and relationships within the data, ensuring that the clusters are reliable and coherent (Jeatrakul et al., 2010). Thirdly, data cleaning reduces the dimensionality of the dataset, simplifying its complexity (Corrales et al., 2018). Because of this, the clustering process becomes

more computationally efficient and easier to interpret. Fourthly, language standardization is another critical outcome of data cleaning. Techniques such as stemming, lemmatizing and removing project-specific vocabulary standardize the language within the dataset. This standardization is invaluable for identifying common themes and concepts across activities, even when expressed differently. As a result, when using vectorization that treats text as numerical data, the thorough data-cleaning procedure is significant when developing meaningful data clustering.

3.2.1. Part of Speech

First, all separate words in a sentence need to be tokenized. This is the first step in identifying the different parts of speech. This tokenization is a critical part of preforming NLP, which converts sentences into smaller tokens, with each of these tokens being separate words (Hassler & Fliedl, 2006). Building upon the tokenization process, the next step involves assigning parts of speech (PoS) tags to each tokenized word. This is accomplished by applying the Natural Language Toolkit (NLTK) library, specifically the PoS_tag function.

3.2.2. Text between brackets and punctuation marks

In written text, it can be seen that additional information is enclosed between the bracket punctuation marks within a sentence (Arya, 2018). Therefore, this data cleaning step aims to eliminate all text enclosed within brackets.

3.2.3. Stemming and Lemmatization

The stemming of words is the process of standardising words by reducing a word to its root or base form (Rianto et al., 2021). This will lower the number of variants of one word in the data. The method of stemming is especially helpful in natural language processing tasks, as it mitigates the challenges that different variants of one word bring. Using the PorterStemmer method, the words will be reduced to their root/base.

Lemmatization is closely related to the stemming process. While stemming reduces a word to its stem, lemmatization removes unnecessary word endings (Samir & Lahbib, 2018). Furthermore, lemmatization is described as a more precise method than stemming (Balakrishnan & Ethel, 2014). However, the article of Balakrishnan & Ethel (2014) indicates that the differences are insignificant. Therefore, both will be tested in this research.

3.2.4. Translation of Dutch words

One of the things that was noticed in the dataset is that the language is not consistently English. Since this can cause some problems regarding the clustering, the decision is made to remove the non-English words. In order to do this, a word detection program is needed that will detect singular words and not only the overall language of the document. This language detection per word comes with some challenges. Some words are used in multiple languages (e.g. die has different meanings in Dutch and English). Therefore, this data preparation method will not have a perfect score, but removing the largest part of the non-English (Dutch) words is beneficial. Since different language detection methods work differently, some will better detect singular words than others. In order to find out what works best, Langdetect, langid, textblob, and spacy-langdetect have been used. After manually evaluating the results of each of these libraries, it was found that spacy-langdetect works best for detecting singular words in another language.

3.2.5. Project specific terms

Due to the non-project-specific approach, the aforementioned data cleansing methods are universally applicable. However, after this universally applicable data cleaning, much unnecessary project-specific information remains in the data set. This unnecessary project-specific information can be seen as noise in the clustering process. Given that the clustering process gets increasingly challenging as noise levels rise, this noise may reduce clustering accuracy. Therefore, it is crucial to eliminate project-specific terms from the data. Systematic categories are defined to facilitate some structure in defining these project-specific terms. These categories include words featuring digits, directions, components, count numbers, and a residual category termed "other terms." This residual category includes the words that call for removing the data but do not fit into another category. The identification and removal of these residual terms will be executed by initially running the removal of words from the other categories. A thorough data follow-up check is needed to check if additional words need removal.

3.2.6. Testing data preparation

These different ways of data preparation can be used individually or together in different formats. All the different data preparation steps will be tested individually to determine what would work best. Next, combinations of preparation methods and different orders of all data preparations are tested. The data preparation method with the highest-scoring cluster similarity will be used for the final clustering.

3.3. Naming clusters

Within the K-means algorithm, the clusters will be indicated by numbers. However, when this is used for the labelling of a new planning, this is not the most optimal since a number does not give insight into what the cluster entails. Therefore, this part will cover giving a name to the made clusters. This process can be executed through various methods, primarily falling into two main categories: manual and automated labelling. Within this project, the approaches tested include manual labelling, labelling based on the most prevalent words within each cluster, and labelling based on the most frequently mentioned activities within each cluster. The following sections will provide detailed explanations of these methods. Last, every method is concluded by an explanation when this method would best be used.

3.3.1. Manual naming clusters

As the name indicates, the manual labelling method is executed through direct human intervention involving a non-automated process. This approach consists of looking through the clusters made and writing down the overarching theme of each cluster. This approach offers several advantages and disadvantages. The main advantage is that precision within each cluster can be given, all the activities are analysed, and an overarching theme can be found that does not have to be the main activity (or words) in that cluster. Furthermore, manual labelling is better customizable to project-specific needs. However, this method has its downsides; it is time-consuming, introduces subjectivity and potential biases, and has scalability challenges. This last disadvantage is a primary issue for this research. Since this research has over 300 categories, it becomes hard to analyse manually if clusters overlap. Furthermore, it is also not scalable to another project since it requires input, skills and availability of a human. Therefore, this method would only be used, if the dataset is small and comprehensible. Since this will not be the case for this research, the method will not be used.

3.3.2. Naming of clusters based on the most prevalent words

The next approach is to use the most prevalent words in every cluster as the cluster label. For this, an algorithm first identifies these words and then generates a label per cluster. It is important to note that the algorithm used does not include stop words in generating these labels. This approach comes with several advantages. Firstly, it is highly efficient and scalable, making it well-suited for processing large datasets. It also offers consistency in labelling, as it adheres to predefined rules, reducing the potential for human subjectivity and biases. However, there are notable downsides to this method. One key drawback is the potential lack of context and nuanced understanding of the labels. Especially when the most prevalent words are used, it could be the case that these words do not form a coherent label together as a label. Therefore, using this method with an extensive dataset, with different factors is not viable.

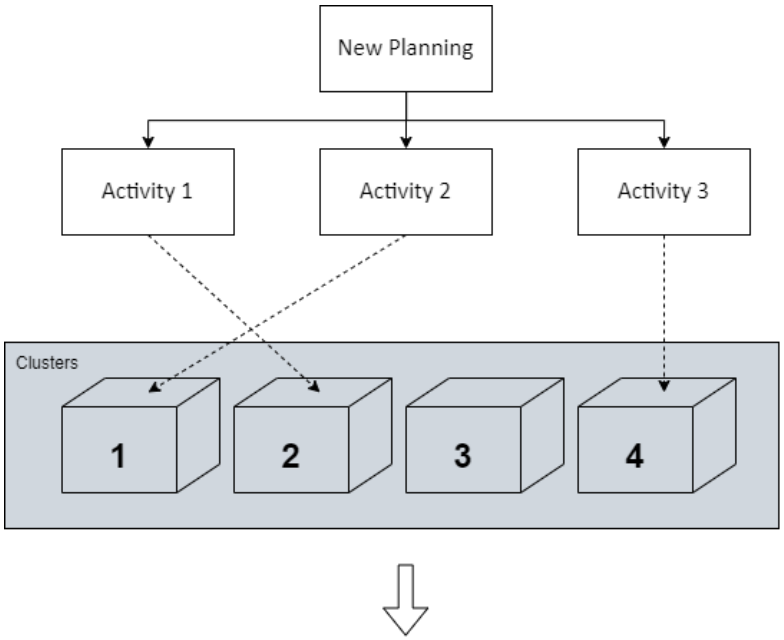
3.3.3. Naming of clusters based on the most frequent activity

The last option is to label the cluster based on the most frequent activity. This approach is also automated with the use of an algorithm. This algorithm analyses all the cluster activities and chooses the most common activity. The advantages are comparable to the advantages in Section 3.3.2. An additional advantage is that this approach will generate a coherent label. However, there are also some disadvantages. However, it has drawbacks, similar to the previous approach, such as the potential lack of context and nuanced understanding of labels. Unlike the previous section, a unique disadvantage is that it does not consider the other activities in a cluster, which means that the second most frequent activity when generating labels is not considered. In a extensive dataset, this is the most viable method. Therefore, this method will be explored in this thesis, while still taking into account the disadvantages of tis method.

3.4. Labelling of new planning activities

The previous steps helped make the initial clusters the basis for labelling new planning. To facilitate understanding, Figure 12 visually represents the clusters (labelled as clusters 1 to 4 in this instance) in the form of boxes. These boxes represent the clusters generated through a clustering algorithm, as elaborated in Section 3.1.

The primary objective of this research is to assess the planning aspects of two different projects. In this phase, a new project is introduced, and the challenge is to appropriately assign its activities to the relevant box (cluster). This involves ensuring that the activities align with those already in the box from the initial planning. The process of placing these new planning activities into the boxes is called the labelling process. In essence, every new activity will get a label corresponding to the box it is placed in.



Activity	Clusters
1	2
2	1
3	4

Figure 12 Explanation of the clusters and labelling process

In order to facilitate this labelling process, a code is written in Python. The main focus of this code is to place the activities in the correct cluster and ensure this cluster is labelled to this activity. The complete method of this code can be seen in Figure 13. This methodology will explain how a new project planning is labelled. Initial clusters are formed through K-means clustering, and their labels are loaded. The new dataset is processed by applying a vectorizer and reducing dimensionality, aligning with the techniques used during clustering. The pre-trained K-Means model predicts cluster labels for the new, dimensionality-reduced data. These predicted labels are then assigned to the new dataset, indicating the appropriate cluster for each activity. The resulting labelled dataset is saved, in which every activity is labelled to a corresponding cluster.

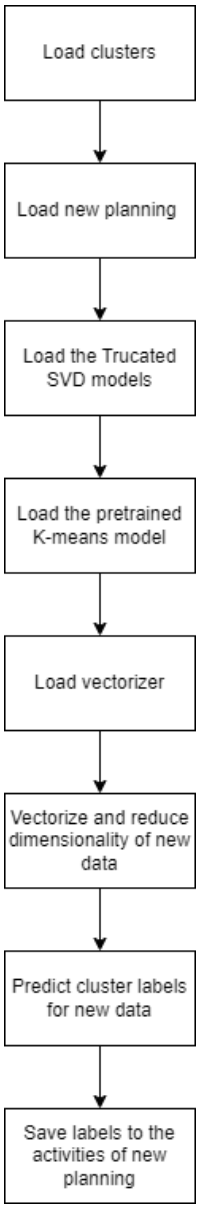


Figure 13 Labelling methodology

3.5. Validation of clusters and labelling algorithm

The labelling process is applied to two different project plans to validate the clusters and evaluate the efficacy of the labelling algorithm. These plans vary based on the context and project phase, allowing for a thorough examination of the algorithm's accuracy and areas for improvement. For both projects, the validation process follows a set methodology.

The initial step involves data preparation, following the same procedures applied to the basis planning used in cluster creation. Tasks include translating each plan into Dutch and defining project-specific abbreviations and terms. These preparatory steps are needed to integrate the planning into the labelling algorithm. Following data preparation, the imported data goes through to the labelling algorithm, which assigns each activity to the best-fitting cluster. Following allocation, a comparison of outcomes is done. Newly assigned activities are reviewed concerning the cluster title and existing activities within that cluster. This analysis aims to reveal the algorithm's accuracy in labelling activities while providing insights into the clusters' cohesion and consistency.

This method of validation not only evaluates the algorithm's performance but identifies potential areas for improvement in the data preparation and clustering processes. The differences in context and phase in the two projects provide a more nuanced perspective on the algorithm's adaptability and effectiveness in various scenarios.

3.6. Methodology conclusion

This chapter shows the steps taken to first cluster a project planning and later label the new planning activities into these made clusters. To give a complete overview of the process, Figure 14 was made. The next chapter of this thesis will explain the results of the steps described in this chapter.

This chapter carefully describes the steps needed to cluster a project planning, followed by labelling the newly introduced activities. Figure 14 captures the whole procedure, offering a thorough perspective of the path made. The following chapter in this thesis will look into the consequences of the methods described. The clustering process is an attempt in which the objective is not only to cluster project planning elements but also to designate the clusters with a relevant label. Figure 14 provides a visual synthesis of the transformation from raw project data to separated clusters. Each process stage has been carefully documented, providing readers with a clear and informative representation of the methods used. The intended clustering of project planning aspects becomes a fundamental step, laying the groundwork for the following labelling phase. In the results chapter, As we go into the following chapter, the emphasis will shift to assessing the outcomes and insights generated from the clustering and labelling techniques. This next part sheds light on organised clusters' relevance and function in improving knowledge between different project planning. The outcomes will be described, providing links between the basic actions outlined in this chapter and the broader implications for project management.

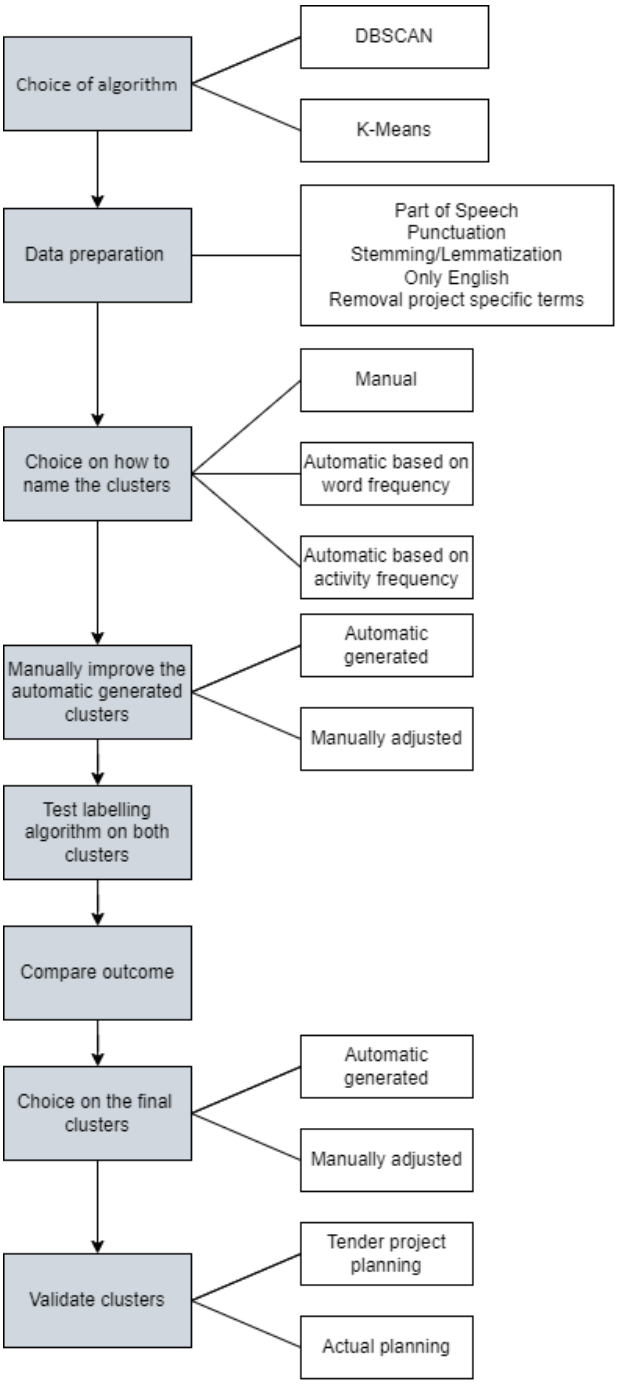


Figure 14 Complete overview of methodology

4. Results

This chapter will present the outcomes and findings derived from applying our clustering and labelling methodology to evaluate the planning aspects of two distinct projects. The preceding chapter detailed the development of initial clusters, each representative of a specific type of activity. To do this, the data used for this will be explained. Next, the chapter will describe how the different clusters are formed based on the algorithm choice, data preparation and manual changes. Furthermore, it describes how the different clusters are named and eventually explains how a new planning can be labelled according to the corresponding clusters.

4.1. Data extraction

The data should be gathered and prepared before the different algorithms can be tested for this research. For this thesis, the data used is provided by Count & Cooper. The data used in this thesis is stored within a Common Data Environment (CDE), a centralized cloud-based digital platform employed by Count & Cooper. Within the CDE, project information is stored, managed, and exchanged in a common environment (Werbrouck et al., 2019). SQL is used to access the data. This system allows users to create and oversee database tables and add, edit, and remove records. Furthermore, SQL allows users to execute queries that extract specific data from the database. The database systems supporting the CDE keep several linked data tables, making it easier to manage related information in different parts of a project and organisation.

An SQL system consists of several components: SQL tables, SQL statements, and stored procedures. A SQL table is a basic element of a relational database. Like an Excel table, an SQL table consists of rows and columns. An SQL statement, also referred to as an SQL query, can be seen as instructions given to the database to extract information. Software developers use different SQL language components to create SQL statements. The parts that make up a proper SQL statement, such as identifiers, variables, and search conditions, are known as SQL language elements. Lastly, a stored procedure is a collection of SQL statements used to improve the efficiency and performance of the database. To get the data from the CDE, a SQL query is used. In this thesis, the data from the Veenix Project is used. VeenIX is an infrastructural project for the widening and deepening of the highway A9 between Badhoevedorp and Holendrecht. Within this project, Count & Cooper is responsible for the project management. All the necessary data can be obtained via the CDE of Count & Cooper, including the planning activities of different plannings and the actual lead times. The complete Veenix planning with all activities can be accessed via this SQL query:

```
select
    act.[Id],
    act.[Name],
    act.[Type],
    actCode.[Description] as [Label]
from primavera.Activity act
left join primavera.Activity_Code act_code
on act.[objectId] = act_code.[ActivityObjectId]
left join primavera.ActivityCode actCode
on act_code.[ValueObjectId] = actCode.[objectId]
where act.[PlanningName] = 'Detailed Planning' and [TypeObjectId] = 219
--*VX | 16 - Work Discipline
```

Listing 1 SQL query to retrieve data

The data retrieved from this query is the complete detailed planning from Veenix. Detailed planning means that all the activities are in there, not the general overarching ones. The data is structured in the following way. It consists of four columns: the ID-code, Name, Type, and Label of every activity. In Table 5, a selection of the data can be seen.

Table 5 Selection of data project Veenix

ID	Name	Type	Label
DC93280	Start KW084 Northside Deepcut (27-3-2023)	Start Milestone	Civil works
ST1873471	Temporary Polderweg West in use	Finish Milestone	Roadworks
L3.15110	Perform UXO-detection KW002 (AB02 and North) L3	Level of Effort	UXO
STXML_5550	Replacing existing bearing blocks KW001 South East (6 axis' & 3 wks/axis) (AFTER BVP1)	WBS Summary	Steelworks
E1828731	Install cable LIANDER 1040 MS 073a	Task Dependent	Utilities
TSS-OD-5	Review DO Temporary Seepage Screen Oude Dorp based on adjusted alignment by-pass (Calculations by Fugro)	Task Dependent	Design
A3010585	BLVC Factsheet for Site Road Works KW009: Send 1 month in advance to municipality	Task Dependent	Delivery packages
A3010745	BLVC Factsheet for Site Road Works KW011 Phase 1: Make	Task Dependent	Delivery packages

Below, all these columns are explained:

1. ID: Every activity has an ID-code; this code is unique to the project so no other activity can have this code. The ID-code of an activity is always built in the same way. Namely, it starts with a letter (or letters) and is followed by several digits. When a new activity is made within a project, there are three options:
    1.1. When initiating the creation of a new activity while another activity is selected, the system automatically duplicates the initial letter(s) of the activity name and increments the numerical value by 10 (in cases where the incremented value results in an already existing ID-code, an additional increment of 10 is applied).
    1.2. Conversely, when creating a new activity from a context where no activity is currently selected, the system employs the letter 'A' as the initial character and again increments the numerical value by 10 (similarly, in cases of a conflict, an additional increment of 10 is applied).
    1.3. When an activity is copied, Primavera duplicates the initial letter(s) and appends an increment of 10 to the numerical value (again, when the system encounters a conflict, an additional increment of 10 is employed).

Option two states that the letter A is applied when a new activity is made. However, the other letters also have a meaning behind them. This logic is shown in Appendix B. The outcome of this logic is also validated with the head planner of Veenix to confirm that the explanation of each letter is correct.

2. Name: This is the text that a planner fills in when making a planning. There are no rules for this field; however, a similar structure between the same kinds of activities can be seen in the two lowest activities in Table 5.

3. Type: In the dataset, there are five different types, namely Start Milestone, Finish milestone, Level of Effort, WBS summary, and Task dependent. The Start and Finish milestones are (as the name indicates) the beginning and end of a milestone. A milestone is a one-time event that represents a specific moment rather than an activity with a set duration. The Level of Effort summarises the corresponding Start and Finish milestones. The WBS Summary is an automatically indicated summary made in primavera from the WBS. Lastly, Task Dependent is the type of activity that this thesis will focus on. The activities with the Task Dependent label are the activities that will be done during the execution phase.

4. Label: This column indicates the previously made labels by planners. These are manually made to fulfil the needs of planners. There is no logic behind the size of each label. However, they can give some insight into what each activity entails.

4.2. Multi Criteria Analysis

This chapter will explain how the decision was made and which two algorithms were tested. The MCA is used to ensure that the algorithms' advantages and disadvantages are taken into account. In Section 3.1. these algorithms and the criteria on which they are scored are explained. Below, the outcome of the MCA is given, with an explanation of the scoring of each algorithm.

Table 6 MCA on unsupervised clustering technique

Criteria	K-Means	Mean Shift	DBSCAN	Agglomerative Hierarchical
Quality of Clustering	4	4	5	3
Robustness	3	3	4	4
Scalability	4	2	4	3
Interpretability	4	3	3	4
Ability to Handle Varying Cluster Shapes and Sizes	3	5	5	4
Computation Time	4	3	4	3
Total	22	20	25	21

K-Means:

1. **Quality of Clustering:** K-Means produces relatively accurate clusters by minimizing the variance within clusters. It groups similar activities together reasonably well but might struggle with non-spherical or irregularly shaped clusters. Score: 4
2. **Robustness:** K-Means can be sensitive to the initial placement of centroids, potentially leading to varying results upon different runs. However, running K-Means multiple times with different initializations and choosing the best result can enhance its robustness. Score: 3
3. **Scalability:** K-Means is relatively scalable and can handle large datasets efficiently. Its time complexity is linear with respect to the number of data points and clusters. Score: 4
4. **Interpretability:** K-Means provides easily interpretable clusters where each data point belongs to the cluster with the closest centroid. This simplicity contributes to good interpretability. Score: 4
5. **Ability to Handle Varying Cluster Shapes and Sizes:** K-Means assumes spherical clusters of similar sizes, which might limit its performance in handling varying shapes and sizes. Score: 3
6. **Computation Time:** K-Means has a reasonable computation time, especially with many activities. Its convergence is generally fast but can be sensitive to the number of clusters. Score: 4

DBSCAN:

1. **Quality of Clustering:** DBSCAN excels in grouping similar activities together, particularly in dense regions. It can discover clusters of arbitrary shapes, enhancing the quality of clustering. Score: 5
2. **Robustness:** DBSCAN is relatively robust since it does not depend on random initializations. Its clustering is determined by the data density and distance thresholds. Score: 4
3. **Scalability:** DBSCAN's scalability can degrade in high-dimensional spaces due to the curse of dimensionality. However, it is generally efficient and effective for large datasets with varying cluster sizes. Score: 4
4. **Interpretability:** DBSCAN's density-based nature can make interpreting less intuitive than centroid-based methods like K-Means. Interpreting the significance of density parameters might be challenging. Score: 3
5. **Ability to Handle Varying Cluster Shapes and Sizes:** DBSCAN excels in handling varying shapes and sizes due to its density-based approach. It adapts well to clusters with irregular shapes and sizes. Score: 5
6. **Computation Time:** DBSCAN's computation time is generally reasonable, but it might take longer in higher-dimensional spaces or for very large datasets. The algorithm complexity depends on the number of data points. Score: 4

Agglomerative Hierarchical Clustering:

1. **Quality of Clustering:** Agglomerative Hierarchical Clustering produces reasonable quality clusters by iteratively merging similar data points. However, its quality might vary based on the chosen linkage criterion. Score: 3
2. **Robustness:** The robustness of Agglomerative Hierarchical Clustering depends on the chosen linkage method. Different linkage criteria can lead to different results, impacting its overall robustness. Score: 4
3. **Scalability:** Agglomerative Hierarchical Clustering's time and space complexity can be high, making it less suitable for large datasets. It might suffer from scalability issues. Score: 3
4. **Interpretability:** Agglomerative Hierarchical Clustering provides a dendrogram that can be used to interpret hierarchical relationships between clusters. However, interpreting the dendrogram might be complex for large datasets. Score: 4

5. **Ability to Handle Varying Cluster Shapes and Sizes:** Agglomerative Hierarchical Clustering can handle varying shapes and sizes to some extent. However, its performance might vary based on the linkage method and the dataset. Score: 4
6. **Computation Time:** Agglomerative Hierarchical Clustering's computation time can be relatively high due to its hierarchical nature, making it less suitable for large datasets. Score: 3

Mean-Shift:

1. **Quality of Clustering:** Mean-Shift excels in grouping similar activities together in meaningful clusters. Its kernel density estimation approach identifies modes in the data distribution, resulting in accurate clustering. Score: 4
2. **Robustness:** Mean-Shift is robust since its results are determined by the underlying data distribution rather than random initialization. It tends to produce consistent results across runs. Score: 3
3. **Scalability:** Mean-Shift's scalability might be limited in high-dimensional spaces or with large datasets. Its time complexity depends on the number of data points and the bandwidth parameter. Score: 2
4. **Interpretability:** Mean-Shift's interpretability is generally good, as its clusters correspond to modes in the data distribution. However, determining the bandwidth parameter might require careful consideration. Score: 3
5. **Ability to Handle Varying Cluster Shapes and Sizes:** Mean-Shift adapts well to varying cluster shapes and sizes due to its density-based nature. It can capture clusters with irregular shapes and sizes. Score: 5
6. **Computation Time:** Mean-Shift's computation time can be moderate but slower for large datasets or high-dimensional spaces. The choice of bandwidth parameter also influences computation time. Score: 3

Based on the MCA, the summarized results are:

- **K-Means**: Scores well in clustering quality, scalability, interpretability, and computation time but less well in handling varying cluster shapes and sizes.
- **Shift means**: Adaptable to various cluster shapes, excelling in non-spherical distributions. However, it is less suitable for high-dimensional data and scalability, making it a balanced choice for specific clustering tasks.
- **DBSCAN**: Excels in clustering quality, robustness, ability to handle varying cluster shapes and sizes, and computation time. It scores lower in interpretability due to its density-based nature.
- **Agglomerative Hierarchical Clustering**: Strengths in robustness, interpretability, and ability to handle varying cluster shapes and sizes. However, it scores lower in quality of clustering and scalability.

The two highest scoring algorithms (DBSCAN and K-means) will be tested to determine which algorithm can be best used in this research. This testing process will be done as follows. First, a test dataset will be made. This test dataset will be made with activities in the same format as those used for the final clustering algorithm. However, since this step of the process focuses on the usability of the algorithm and not the data, a smaller sample of the data will be tested. This way, the focus will lay on the performance of the algorithm.

4.3. Testing of DBSCAN and K-means

The general methodology for the use of the different algorithms is explained within the methodology. This section will explain the algorithm-specific aspects of the two highest scoring algorithms (DBSCAN and K-Means). First, a general overview of the steps has been given in the methodology. This chapter highlight the specifics of the testing of both algorithms. The whole process can be seen in Figure15.

First, the three steps of the DBSCAN after the truncated SVD will be explained. As mentioned in the literature review, when using the DBSCAN, the Eps and Minpts need to be defined. A rule that is used for this is that the starting point for Minpts is two times the dimension, which would mean, in this case, four (Murugesan et al., 2021; Naik Gaonkar & Sawant, 2013; Sander et al., 1998). However, this only serves as a starting point; adjustments to this parameter by hand are needed to find the most optimal outcome. When the Minpts are determined, the Eps can also be determined. There is no general rule of thumb to estimate the Eps. However, one general rule is that the higher the value of Minpts, the higher the Eps should be (Hossain et al., 2021). This also creates some difficulties in finding a general approach regarding the clustering. A benefit, however, is that the algorithm is highly adjustable for different data formats (Luchi et al., 2019).

Since conducting the DBSCAN method is completely elaborated on, the K-Means clustering will be explained in this paragraph. After reducing dimensions with the truncated SVD, the K needs to be defined for the K-Means clustering. As mentioned in this thesis, the research will focus on the optimal number of clusters, which will not be defined beforehand. Within the K-means algorithm, this is accomplished by calculating and plotting silhouette scores, which measure clustering quality. Silhouette scores help evaluate the cohesion and separation of clusters, with higher scores indicating more distinct and well-defined clusters. The silhouette score represents how well data points are assigned to clusters, considering both the tightness of each cluster and the separation between clusters. By generating a plot of silhouette scores against the number of clusters, the relationship between clustering quality and the number of clusters can be seen. Choosing the right number of clusters is a critical decision in clustering analysis, and silhouette scores provide an objective criterion to make an informed choice. When the K with the highest silhouette score is defined, the K-means clustering can be utilized. Therefore, the K-means will be imported into the Python script. In the end, the clustering can also be visualized based on the two-dimensional plot. Since the points in the plot will be the same as for the DBSCAN, the cluster quality can be analysed straightforwardly.

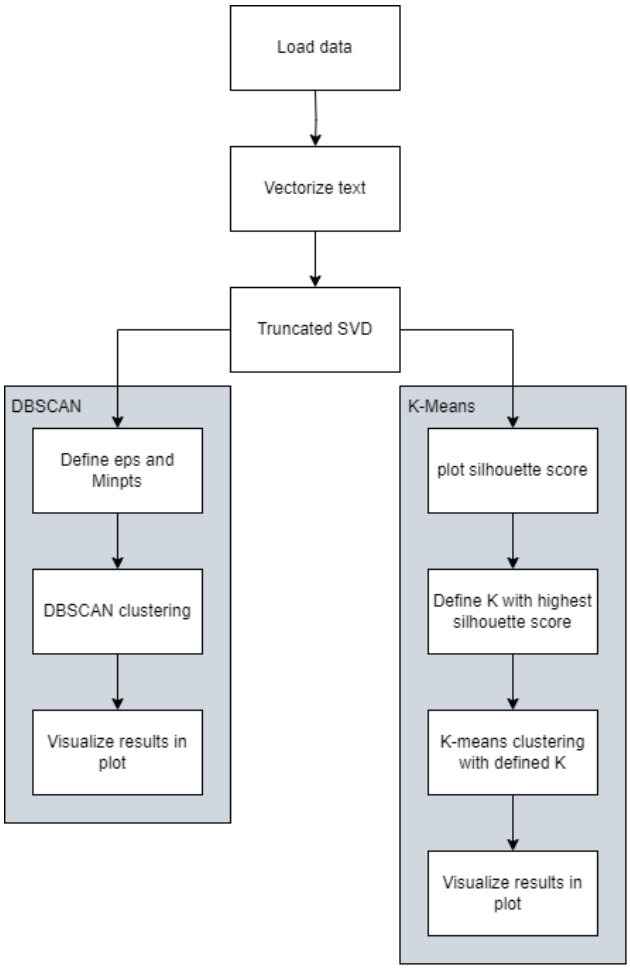


Figure 15 Method DBSCAN and K-Means

After this general overview of how both algorithms work, the remainder of this section will decide between the DBSCAN and K-Means clustering algorithms. The methodology already explained the steps of each of these algorithms. To find out which algorithm works best for this research, both algorithms were tested on the generated data from Section 4.1. Data extraction. First, the DBSCAN algorithm is tested since this algorithm scored the highest in the MCA performed in Section 3.1. A sample data set of the Veenix planning will be used to test which algorithm will be chosen.

As explained in the methodology, the DBSCAN and K-means algorithms overlap in the first few steps. Within these steps, the decision on the vectorizer needed to be made. From the methodology, the two possible methods were Count Vectorizer and TF-IDF. When both of these methods were tested together with the whole testing of the algorithms, TF-IDF came out as the better working one. This is because this vectorizer is more consistent with the varied word counts of every activity. In the dataset, each activity varies in word count. TF-IDF normalizes the term frequencies based on the length of each activity, ensuring that the analysis is not skewed by the differing lengths of the documents (Roelleke & Wang, 2008). This was also seen when testing both of the methods multiple times.

4.3.1. DBSCAN

Listing 9 shows the testing of the DBSCAN algorithm. In this section, the choices made within the algorithm are explained. The vectorizer used is, as explained above, the TF-IDF vectorizer. Furthermore, to reduce the dimensionalities of the data, the TrucatedSVD is used for the DBSCAN-specific choices, namely the Eps and Minpts. As explained in the methodology, the rule of thumb regarding the start test point of Minpts is two times the dimension. This guideline is a starting point for testing the different options of Eps and Minpts. Several options are used to find out what different Eps and Minpts do to the algorithm. Generally, a test of 10% of the Minpts is used. It results in only two clusters for the DBSCAN method with Minpts = 4 and eps = 0,4. While testing multiple Minpts and Eps, it was discovered that the tuning process is quite difficult. Therefore, creating good guidelines for this can be challenging. Also, literature does not have one main way of determining Eps and Minpts (Fahim, 2022; Ozge Ozkok, 2017). This has a negative aspect in that when scaling the data, the density within the data also changes. Since the parameters need to change when the data changes, this can cause issues. Also, its use by other people unfamiliar with the algorithm can be hard. Since there is no clear way to adjust the parameters, no clear "rules" can be given to users when using the algorithm. The research of Naik Gaonkar & Sawant (2013) guides the testing of different numbers for Eps since they researched the Eps and Minpoints for large datasets. In the end, this is also the goal of this research.

```
from sklearn.cluster import DBSCAN
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

data = pd.read_csv(r"C:\Users\31648\OneDrive - TU Eindhoven\Documents\2.
Master\Year 2\graduation\6. Labelled activities\all labels\Civil works.csv")

texts = data[" Name"]

vectorizer = TfidfVectorizer()
vectors = vectorizer.fit_transform(texts)

n_components = 2
svd = TruncatedSVD(n_components=n_components)
vectors_reduced = svd.fit_transform(vectors)

# Cluster labels that need to be defined for the DBSCAN
eps = 0.4  # Distance threshold can be adjusted
Minpts = 4  # Minimum number of samples for a core point can be adjusted
dbscan = DBSCAN(eps=0.4, Minpts=4)
cluster_labels = dbscan.fit_predict(vectors_reduced)

# Create a scatter plot of the reduced vectors with colours based on cluster
labels, this is used to visualize how well the clustering is done
plt.figure(figsize=(10, 6))
for label in set(cluster_labels):
    subset = vectors_reduced[cluster_labels == label]
```

```
    plt.scatter(subset[:, 0], subset[:, 1], label=f'Cluster {label}',
marker='o')
plt.title("TF-IDF Matrix Visualization with DBSCAN Clustering")
plt.xlabel("Component 1")
plt.ylabel("Component 2")
plt.legend()
plt.show()
```

Listing 2 DBSCAN algorithm testing

A visual representation is made after using the code provided in Listing 9. This visualization can be seen in Figure 16. This figure shows the two clusters made, which is not the ideal outcome for this clustering.

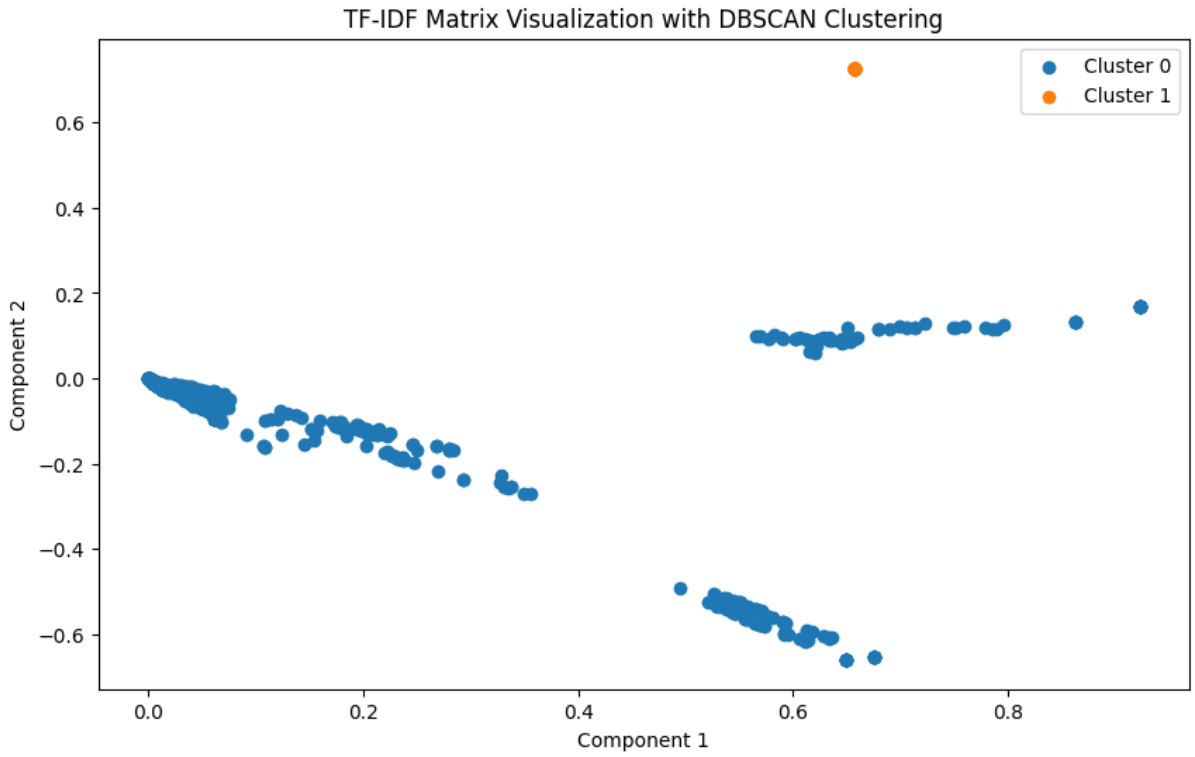


Figure 16 Outcome DBSCAN clustering

4.3.2. K-means

The second algorithm tested is the K-means clustering algorithm. The K-means clustering starts similarly to the DBSCAN algorithm, as the method explains. The specific choices made for the K-means are the specification of the amount of K. To determine this, the silhouette score is used. When using the silhouette score, a decision needs to be made on the range in which the silhouette scores will be plotted. This can be any number between 2 and the amount of activities. The range starts at 2 because this is the minimum amount of clusters. Since 1 is not a cluster, all the activities will be in the same cluster. The max is the amount of activities in the dataset since activities can not be split over multiple clusters. As seen in Listing 10, the range for this sample data is set at 2 to 50.

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from sklearn.metrics import silhouette_score

data = pd.read_csv(r"C:\Users\31648\OneDrive - TU Eindhoven\Documents\2.
Master\Year 2\graduation\6. Labelled activities\all labels\Civil works.csv")

texts = data[" Name"]

# Handle NaN values by replacing them with an empty string
texts = texts.fillna("")

# Initialize the TfidfVectorizer
vectorizer = TfidfVectorizer()

# Fit and transform the text data
vectors = vectorizer.fit_transform(texts)

# Perform dimensionality reduction using TruncatedSVD
n_components = 2  # Number of dimensions for visualization
svd = TruncatedSVD(n_components=n_components)
vectors_reduced = svd.fit_transform(vectors)

# List to store silhouette scores for different cluster numbers
silhouette_scores = []

# Set a range of cluster numbers to try
min_clusters = 2
max_clusters = 50

# Iterate over different cluster numbers
for num_clusters in range(min_clusters, max_clusters + 1):
    kmeans = KMeans(n_clusters=num_clusters, random_state=0)
    labels = kmeans.fit_predict(vectors_reduced)
    silhouette_avg = silhouette_score(vectors_reduced, labels)
    silhouette_scores.append(silhouette_avg)

# Plot silhouette scores
plt.figure(figsize=(10, 6))
plt.plot(range(min_clusters, max_clusters + 1), silhouette_scores, marker='o')
plt.xlabel('Number of Clusters')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Score for Different Cluster Numbers')
plt.show()
```

Listing 3 Sample K-means clustering till silhouette score

This code generated a graph with all the silhouette scores between 2 and 50 K. As mentioned in section 3.1, this thesis will focus on the optimal number of clusters, which will not be defined beforehand. Within the K-means algorithm, this is accomplished by calculating and plotting silhouette scores, which measure clustering quality. Silhouette scores help evaluate the cohesion and separation of clusters, with higher scores indicating more distinct and well-defined clusters. An empty list named "silhouette_scores" is created to accumulate these scores. The code then specifies a range of cluster numbers to investigate, and this is the only thing that needs to be determined manually. As the code iterates through this range, it applies the K-Means clustering algorithm for each cluster number and computes the corresponding silhouette score. The silhouette score represents how well data points are assigned to clusters, considering both the tightness of each cluster and the separation between clusters. By generating a plot of silhouette scores against the number of clusters, the relationship between clustering quality and the number of clusters can be seen. Choosing the right number of clusters is a critical decision in clustering analysis, and silhouette scores provide an objective criterion to make an informed choice. In Figure 17, the visualization of the silhouette scores can be seen. This figure clearly shows that the silhouette score is the highest for four clusters. This will be used in the next steps of the K-means algorithm.

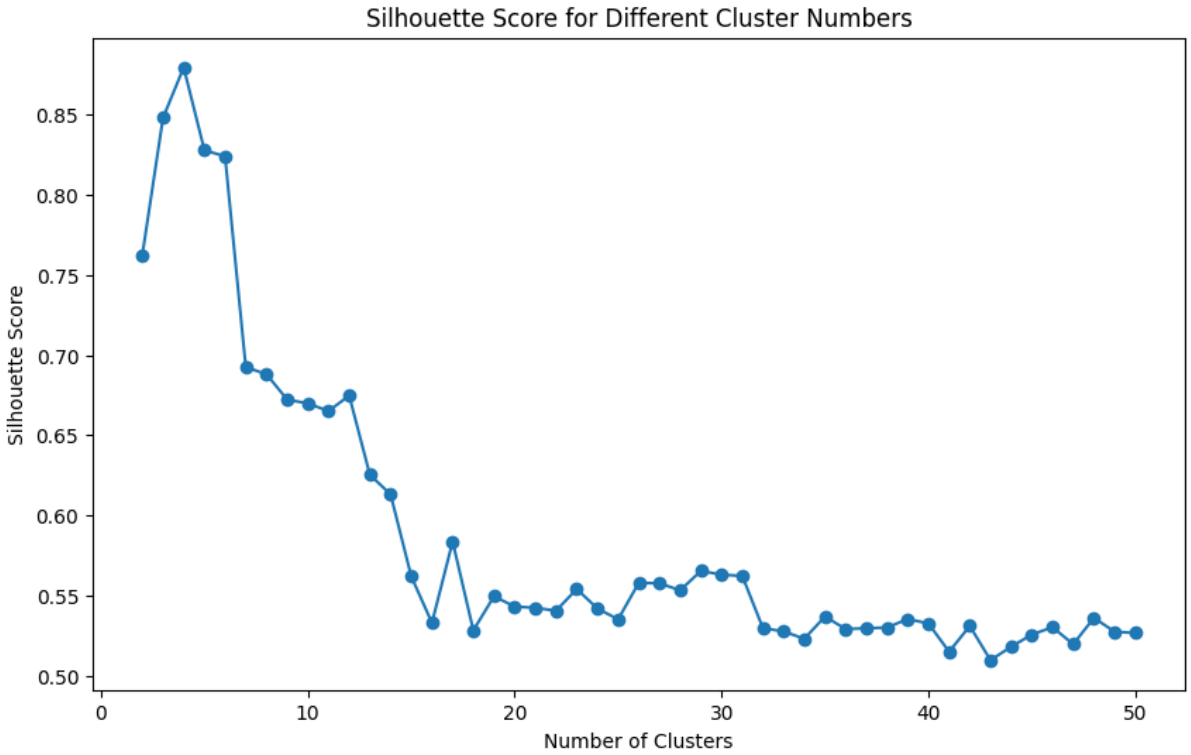


Figure 17 Silhouette score clusters sample data

Figure 17 shows that the specific K (clusters) amount is set at four. The next part of the code, shown in Listing 11, initializes the K-Means clustering algorithm, imported from a library. After this, the code fits the K-Means model to the reduced-dimensional data, "vectors_reduced," and assigns each data point to one of the four clusters using the "fit_predict" method. As a result, the "cluster_labels" variable contains cluster assignments for each data point. This last part of the code is the visualization of the clusters. The result of this can be seen in Figure 18. As can be seen, the four clusters previously made are shown in this figure.

```
# Apply K-Means clustering with a desired number of clusters

n_clusters = 4  # from best silhouette score
kmeans = KMeans(n_clusters=n_clusters)
cluster_labels = kmeans.fit_predict(vectors_reduced)

# Create a scatter plot of the reduced vectors with colors based on cluster
labels
plt.figure(figsize=(10, 6))
for label in set(cluster_labels):
    subset = vectors_reduced[cluster_labels == label]
    plt.scatter(subset[:, 0], subset[:, 1], label=f'Cluster {label}',
marker='o')
plt.title("TF-IDF Matrix Visualization with K-Means Clustering")
plt.xlabel("Component 1")
plt.ylabel("Component 2")
plt.legend()
plt.show()
```

Listing 4 Sample K-means clustering after silhouette score

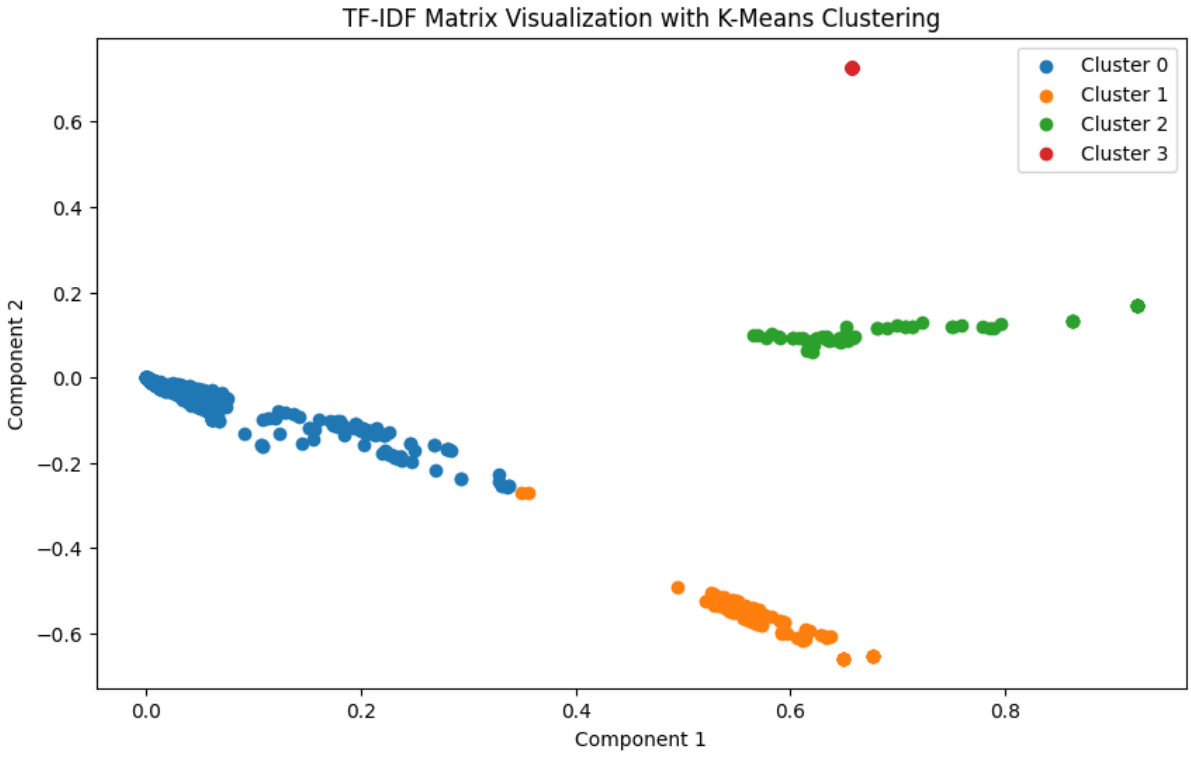


Figure 18 Outcome K-means clustering sample data

4.3.3. Final choice

After testing both algorithms on a sample data set, some conclusions can be made regarding the outcome and use of the algorithms. First of all, the use of the algorithms will be discussed. Both of the algorithms have in common that parameters need to be defined. These parameters will alter the effectiveness of the clustering results. When both algorithms were used, it was observed that for the DBSCAN, more insight was needed to develop the right Eps and Minpts since no general rule to determine the best options for both is in place. While the K-means needs to be defined, there is a straightforward rule for this determination. When plotting the silhouette score graph, which K is the highest can be clearly seen. For the outcome, K-means also provides a better clustering.

After comparing the results of the two algorithms in Section 4.3.1. and 4.3.2., it can be seen that the results of the K-Means in 4.3.2. are more accurate. Since the K-Means generated more clusters than the two clusters made with DBSCAN. Therefore, the choice is made to continue with the K-Means. The complete decision of the method is visualised in Figure 19.

In general, when applying the K-means algorithm to the complete dataset instead of the partial dataset, not much of the code changes compared to what is explained in Section 4.3.2. where the K-Means is tested, the final code used for this clustering can be found in Appendix C.

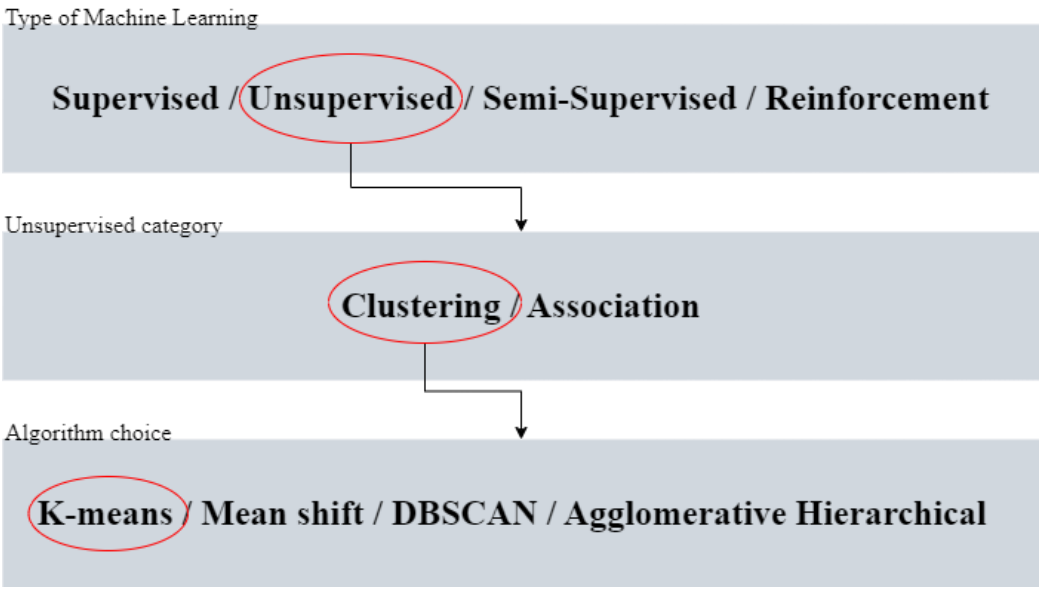


Figure 19 Visualization choice of method

4.4. Outcome data preparation

Since there are multiple ways the data is prepared, it is important to see in what way the preparation methods are beneficial for the result. Section 3.2. explained the different data preparation methods that are suitable for this research. The code used for all these methods can be found in Appendix E. This section will focus on the outcomes of the different data preparation options.

To evaluate the outcomes, the clustering algorithm will be performed for every different data set generated by the data preparation, and the similarity of every cluster made will be generated. The results of the highest scoring data preparation method will be discussed in the next chapter. The code explained in the Methodology chapter is used to generate the results. The final version of the code can be found in Appendix F.

The complete method for the evaluation can be seen in Figure 20. However, some baselines will be set to ensure consistency between the generated results. First of all, the different datasets need to be defined. This will be the original data set and its alterations (PoS, punctuation, lemmatizing/stemming, removal of project-specific terms). The alterations will be done separately to see the influence of each alteration. Lastly, different combinations of the alterations will be done. Next to this, the number of clusters needs to be set. The decision is made to take the best silhouette score of every dataset when the K is set between 200 and 400. This range is set based on the labels and ID codes (from Section 4.1.) assigned to the activities by the planners of Count & Cooper. This range is made after carefully looking at both ways the activities were clustered and how general or specific these clusters were. Furthermore, the reason not to take one set amount of K for all datasets is that it can differ per dataset how high the silhouette score will be, ultimately also influencing the choice of data preparation.

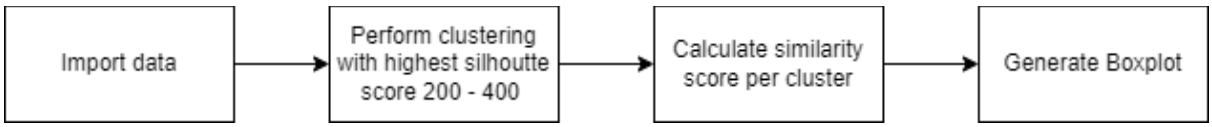


Figure 20 Method for the assessment of the clustering results

As the previous chapter suggests, the data preparation where all the different tools are used, together with a lemmatization tool instead of a stemmer, is the best way to prepare the data to get the highest clustering results regarding the silhouette score. The final score is determined at 0.596. In general, the higher the score, the better the cluster results are. On average, a score above 0.5 is considered reliable (Dalmaijer et al., 2022; Marsani & Acharya, n.d.). This silhouette score can be seen in Figure 21, and a zoomed-in version can be seen in Figure 22. As the figures indicate, the number of clusters with the highest silhouette score is K = 387, meaning that the Silhouette Score is 0.596. The results of the other data preparation options can be found in Appendix G.

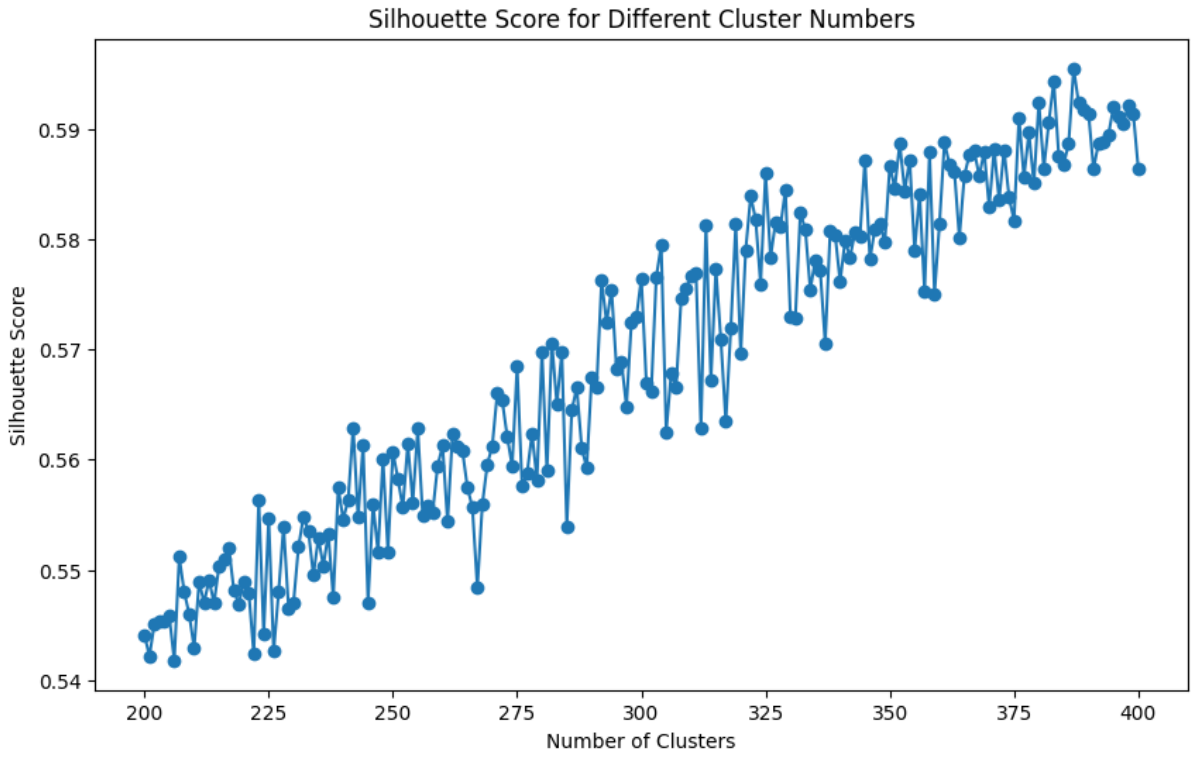


Figure 21 Silhouette score of final clustering

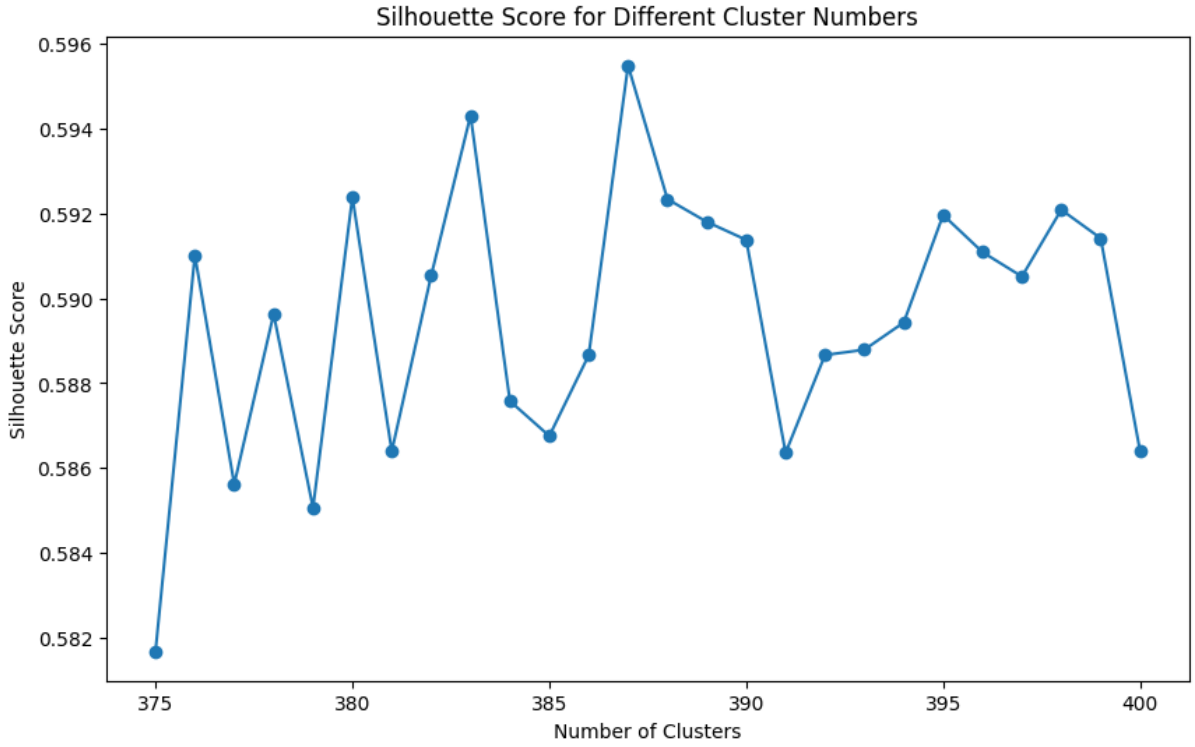


Figure 22 Zoomed in Silhouette score of final clustering

4.5. Naming of clusters

Since the previous sections of the results have determined the final amount of clusters, this section will dive into converting the numerical cluster names to textual cluster names. In Section 3.3 of the methodology, the different methods for the naming of the clusters are explained. This section will explain the testing of these methods and the decisions made. Appendix H can be consulted for a complete overview of the used code.

The different advantages and disadvantages came to light when comparing and testing all of the methods. From trying all these methods, it can be concluded that the manual approach is not usable. The reason for this is that due to the extensive dataset, identifying the labels and patterns within and in between the clusters becomes unfeasible. This conclusion was made after approximately labelling a hundred 378 clusters. During this process, it was discovered that the labelling of approximately one hundred clusters by hand posed challenges distinguishing existing activity groupings to which additional attributions could be made. Next to this, the trying of the methods can also conclude that the labelling based on the most prevalent words is also not feasible. The disadvantage already mentioned in Section 3.3.2, the potential lack of context and nuanced understanding of the labels, was true for this research. When identifying the most prevalent words, the labels became a mix of different activities within those clusters and did not identify the most present themes in every cluster. Therefore, the decision is made to label the activities based on the most frequent activity in every cluster. Overall, this gives a good insight into the most prevalent theme in every cluster. However, it still does have some drawbacks since it will not indicate how much of every cluster identifies with that. To show that the percentage of activities matches the label name (and thus the most prevalent activity), the COUNTIFS function in Excel is used. In the next section, the results of this will be elaborated on.

The final result can be shown since the number of clusters and how to label them are both decided. Appendix I shows a complete overview of every cluster's labels and the corresponding percentage of similarity (named ‘Similarity’ in the table). In this chapter, the results will be further analysed. First, the names of the labels are analysed. Furthermore, the names will be redefined to ensure they are grammatically correct.

When analysing the clustering and labelling results presented in Appendix I, it can be seen that some labels given to the clusters are more straightforward than others. Also, large differences in the similarity can be seen. In this chapter, some labels will be explained in more detail to give insight into the clusters made and their corresponding labels. All labels discussed can be seen in Table 7.

Table 7 Part of clusters and labels for further explanation

Cluster	Label	Similarity
1	Apply layer sand	93%
18	Apply	99%
165	Apply sand SOIL	100%
247	Install duct	5%
321	Wall execution	100%

The first cluster to be discussed is Cluster 1 with the the corresponding label “Apply sand layer”. The activities in this cluster have a relatively high similarity since 93% of the activities in this cluster have the same name as the cluster. However, as Table 7 indicates, other clusters have a similar label. In Table 7, this applies to cluster 165. This has the word soil in caps behind it; therefore, it is put into a separate

cluster. However, these two clusters indicate the same thing. Therefore, these two clusters can be merged into one cluster with the same label name.

Another example that will be touched upon is the clusters with a label only consisting of one word. Based on the “Label” column in Table 7, the name starts the same as for “apply layer sand”, and it can be wrongly concluded that those activities also belong in cluster 1. However, because of the data preparation, it shortened to “Apply”, but the context of these activities differs. When looking at the original name and considering the context indicated there, it can be concluded that this cluster is for applying Argex, a clay pebble. Therefore, all clusters with a corresponding label with only one word will be analysed in depth to see what the new label name should be.

Another example of this is cluster 373. However, for this cluster, the percentage of similarity is also lower. The way to name the label is to look at the activities called “Time” in the adjusted data, see the original name, and find the overarching theme there.

The next remarkable aspect of labelling and clustering is that the clusters where the most frequent activity is only a small percentage of the cluster. An example of this is cluster 321. It is important to note that this cluster also has the lowest percentage. When analysing the cluster, it can be seen that there is a higher overarching theme; therefore, the name will be switched to “Install Duct”. Since this may also be the case for other clusters with a low percentage, all the clusters are in the lower quadrant of Figure 23.

Lastly, an in-depth look will be given to the clusters with a 100% score. If there is only one activity, a critical look will be given if these can be added to another cluster. For the cluster in Table 7, Cluster 321 Wall execution, this is the case.

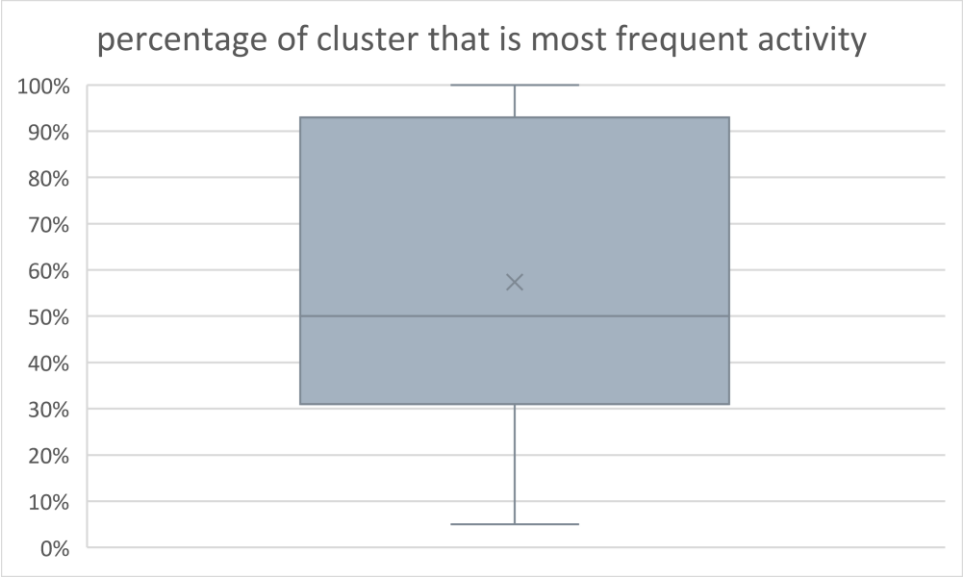


Figure 23 Boxplot of distribution of the percentages of all clusters

4.6. Labelling algorithm

Now that the original algorithm is clustered into the amount of clusters previously defined, it is important to work on the next step. This is the code that will label a new planning into the previously made clusters. Before doing this some aspects of the made clusters in the previous code need to be saved.

```
joblib.dump(kmeans, 'kmeans_model.pkl')
# Save the TF-IDF vectorizer and its vocabulary
joblib.dump(vectorizer, 'tfidf_vectorizer.pkl')
# Save the Truncated SVD model
joblib.dump(svd, 'truncated_svd.pkl')
```

Listing 5 Trained model K-Means

The labelling code looks as followed:

```
import pandas as pd
import joblib
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD

# Load cluster labels from the CSV file
cluster_labels = pd.read_csv('cluster_labels.csv')

# Load new construction planning dataset
new_data = pd.read_csv(r"C:\Users\31648\OneDrive - TU Eindhoven\Documents\2.
Master\Year 2\graduation\6. Labelled activities\all labels\Civil works.csv")

# Load the Truncated SVD model
svd = joblib.load(r"C:\Users\31648\OneDrive - TU Eindhoven\Documents\2.
Master\Year 2\graduation\coding\truncated_svd.pkl")

num_components = svd.n_components

print(f"Number of components in the loaded model: {num_components}")

# Load the pre-trained K-Means model from the training-dataset
kmeans = joblib.load('kmeans_model.pkl')

# Load the TF-IDF vectorizer and its vocabulary
vectorizer = joblib.load('tfidf_vectorizer.pkl')

# Fill missing values in the "Name" column with an empty string
new_data[' Name'].fillna('', inplace=True)

# Transform the "Name" column into TF-IDF vectors using the loaded vectorizer
name_vectors = vectorizer.transform(new_data[' Name'])

# Apply Truncated SVD to match the dimensionality
name_vectors_reduced = svd.transform(name_vectors)
```

```
# Predict cluster labels for the new data
cluster_labels_new = kmeans.predict(name_vectors_reduced)

# Assign cluster labels to the new_data DataFrame
new_data['Cluster'] = cluster_labels_new

# Save or display the new_data DataFrame with cluster labels
new_data.to_csv('new_data_with_clusters.csv', index=False)  # Save to a CSV
print(new_data.head())  # Display the DataFrame
```

Listing 6 Labelling code

In the next chapter, two new projects will be used for testing the labelling of both the automatic generated clusters, as well as the manually changed clusters, discussed in this chapter.

4.7. Result Conclusion

This chapter showed the different choices made to eventually make a clustering and labelling algorithm. Within this chapter, the NLP techniques employed, as well as the ML decisions are explained. In the end, this thesis delve into the unsupervised machine learning clustering by employing the K-Means algorithm. Since this is one of the multiple ways that have been explored in this chapter. This section will highlight all the possible options that could have been made to come to an end result. Therefore, Figure 24 is made. Due to the lack of visibility the figure will be enlarged in Appendix J.

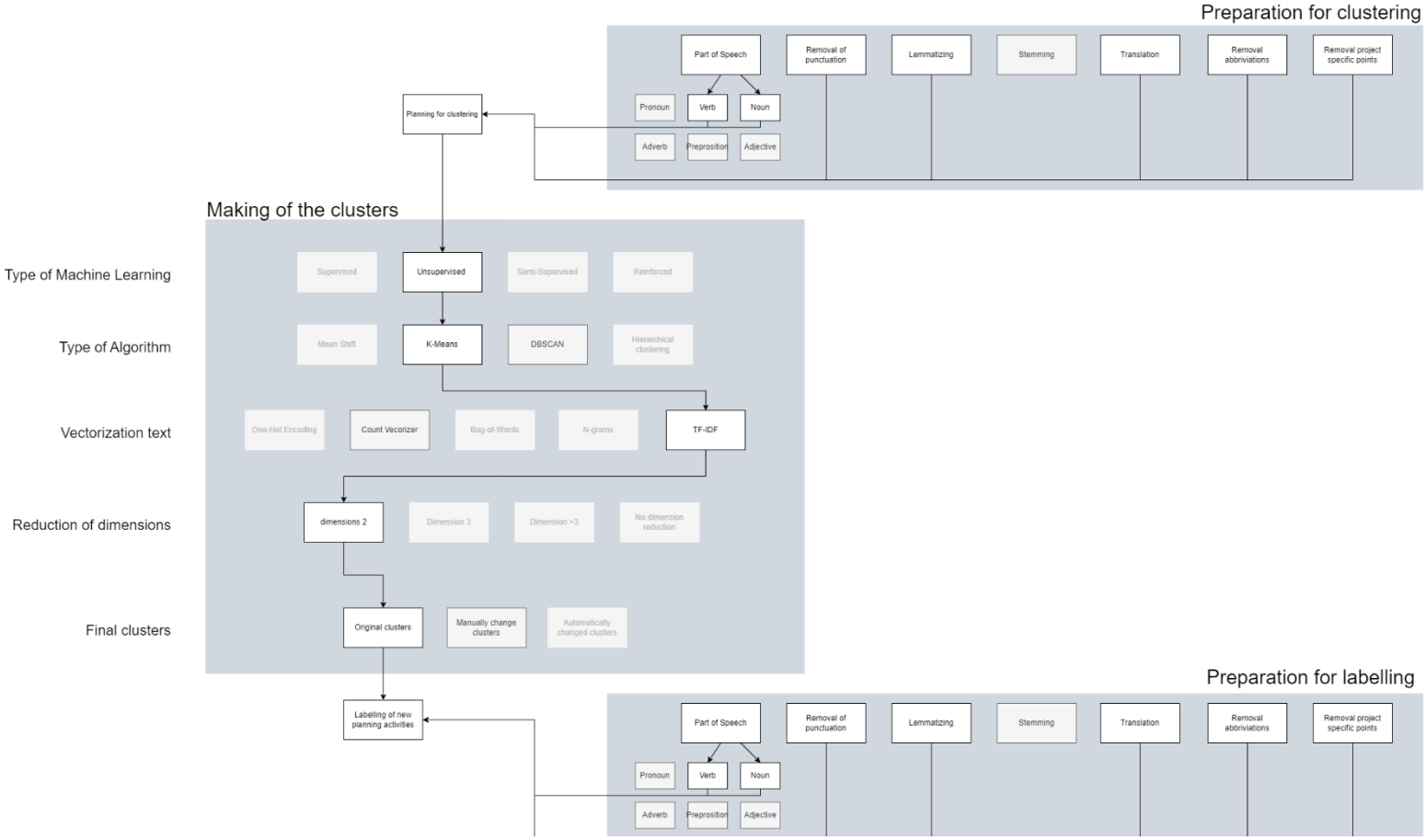


Figure 24 Decision path Results

5. Validation of results

This chapter will validate the results of the clusters presented in Chapter 4, and the labelling code will be validated. This validation is done using two distinct planning scenarios, which were used to assess the effectiveness of the clustering and labelling mechanism. Unfortunately, no other highway project planning was available, so these scenarios were chosen. The first used planning is one of a highway project, but it is different from the original planning because it is a tender planning. The second planning is not a tender planning; however, it is from a waterway project. These two different planning scenarios will give insight into how the labelling algorithm works.

The goal of this chapter is to find out if the approach of unsupervised learning via K-means is suitable for comparing the planning of two projects with each other. For this validation, the clusters made with the K-means model are used. Together with the labelling algorithm shown in the previous chapter, the new projects will be put in for the labelling process. In the end, it can be seen if unsupervised clustering is the right approach to reach the aims, as defined in this thesis's introduction.

5.1. Highway tender planning

The first planning that will be used for the validation is the planning of A15. This is a tender planning, which means that this planning is made for the tender process and not during the official execution of the project. In this section, the data within the planning will be explained. As well as how the labelling of this planning compares to the initially made clusters.

First, the data is explained to give insight into the planning. The tender planning of the A15 consists of an activity name and a type of task, as explained in Section 4.1. As for the original data, only the task-related activities will be chosen to execute this validation. The names of these activities are written down completely in Dutch. Since the clustering is based on English activities, the activities of this highway tender will need to be translated. The described data preparation, used for the original planning, will also be used for this validation. In this data preparation, translating the Dutch words is already a step. Luckily, this code is also able to translate a whole data file. Most of these data preparation steps can be taken without alterations within the code. The steps that require alterations are removing project-specific terms and abbreviations. These must be defined separately for every planning imported into the labelling mechanism. The adjusted code, with the project-specific terms for this tender planning, can be found in Appendix K. The two tables below illustrate this data preparation process. Table 8 shows the names of the original activities in Dutch. Table 9 shows the alterations of these activities after the data preparation methods are employed.

Table 8 Original names of the activities of the A15 planning

Name	**Type**
Verkrijgen percelen t.b.v. aanleg Snelfietsroute SF15	Task Dependent
Validatie voorlopig ontwerp - Geluidsschermen	Task Dependent
Plaatsen dek viaduct - KW37	Task Dependent

Table 9 Adjusted names of the activities of the A15 planning after data preparation

Name	Type
Acquir plot construction Fast Cycle Route	Task Dependent
Validat design Noise barrier	Task Dependent
Plac deck viaduct	Task Dependent

After the data preparation, the data can be entered into the labelling mechanism, shown in Section 4.6 and Appendix F. The mechanism will show, as a result, all the activities of the A15 planning, together with the corresponding cluster. When all the activities are labelled, the outcome can be analysed. Within this validation process, two main outcomes stood out. First, it is noted that the labelling of activities in the new planning primarily relied on verbs as descriptors. As explained in Section 3.2.1, the nouns and verbs of the planning were used for the clustering and labelling. However, when looking at the validation, with the use of new planning, it can be seen that the verbs are mainly used for the clustering instead of the nouns. This choice was attributed to the higher consistency observed among verbs across different planning instances, while nouns tended to exhibit more project-specific variations. The only instances where the nouns were used were when this noun was present a lot in the original planning and this new planning. An example of this is the case for the noun “Sand”, this can be seen in Table 10.

Table 10 Clustering on noun "Sand"

Supply apply cunet sand loop	294
Supply apply sand filling	294
Supply apply sand filling	294
Supply apply sand filling	294

Second, it was observed that activities in the new planning were frequently categorized into clusters with less inherent similarity. In such cases, a high match score derived from the table in Appendix I decreased the likelihood of an activity being labelled in a manner incongruent with the cluster to which it was assigned. However, if an activity is labelled in a cluster with a high inherent similarity, one or only a few activities are labelled in this cluster. The use of clusters with low similarity can be seen in Figure 25. The figure shows the similarity percentages of the used clusters in Appendix H. When comparing the two boxplots, it can be seen that the similarity of the clusters used for labelling are significantly lower than the average of all the clusters.

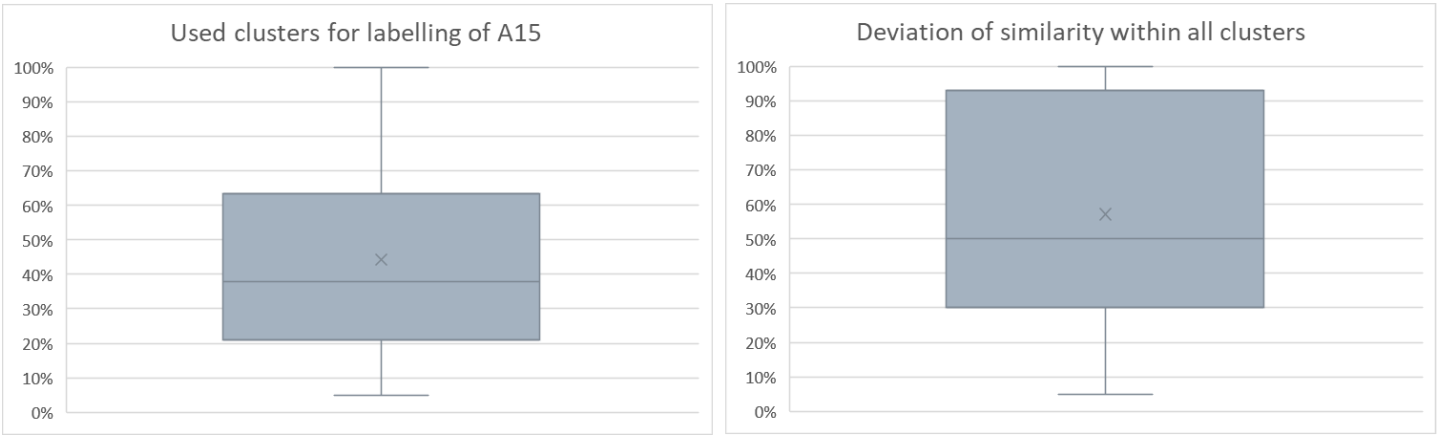


Figure 25 Comparison of used clusters A15 with avarage of all clusters

5.2. Waterway execution planning

The second planning used in this validation chapter is the waterway planning. This specific project is the Twente Canals in the east of the Netherlands. This project has already been executed, so the project planning is more detailed than for a tender. This planning significantly differs from the previous validation and original planning used for clustering. For this validation planning, the data structure is the same as for the basis VeenIX planning and the A15 tender planning. Also, the use of data preparation is the same. Therefore, the project-specific terms removed from this planning can be seen in Appendix K. Therefore, the content of this section will directly dive into the results of labelling these planning activities.

In general, the labelling of the activities of this project was not performed well. Since the two projects are highly different in content, finding common ground between the made clusters and these activities has been difficult. However, some more general conclusions also can be found. First, similar to the previous project planning, verbs played a prominent role in clustering, with activities containing shared verbs being grouped. Again, the rationale behind this was the greater consistency of verbs across diverse planning instances compared to nouns. Second, introducing activities from the new planning was primarily associated with clusters exhibiting less intrinsic similarity. High match scores from the table in Appendix I reduced the likelihood of activities being labelled discordantly within those clusters. Conversely, clusters already comprising diverse activities were more accommodating to integrating new, non-matching activities, driven by the algorithm's adaptability.

5.3. Conclusion on implementation

In general, the validation of both projects shows the same kind of outcomes. Both of the projects are not clustered as efficiently as hoped. The reasons for this will be all summarized in this section. First of all, it starts with the clustering similarity score in general. As explained beforehand, the clustering similarity, based on the silhouette score, is just under 0.6. This has as an influence that still quite some clusters have low similarity within that cluster. When trying to label the activities of a new planning into this, it can be seen that when not a 100% match can be found, these activities will be placed in an cluster where the similarity is already low. Furthermore, when zooming in on how the new activities are clustered, it is based on words that both can be found in the basis planning, and the new planning. What was found out in this research, is that is most of the times the verbs and not the nouns.

In light of the research aim to evaluate project planning effectively across different construction projects, the unsupervised learning approach provides a valuable foundation. It offers insights into commonalities and differences in planning activities, aligning with introducing innovation in planning evaluation processes. However, for a thorough evaluation based on specific activity types, the outcome is insufficient, as described above. Sevral iterations can be made to

6. Discussion

The research done in this thesis has generated two main outcomes: firstly, the clustering of project planning, and secondly, the labelling of two new planning instances within these established clusters. The primary objective of this methodology is to establish a foundation for straightforward comparisons of planning activities between different projects.

The decision-making process concerning the algorithm is discussed initially. The literature review encompassed various machine learning types, and given the pre-existing but strategically unguided labelling of the data, the choice was made to forego supervised learning and proceed with unsupervised machine learning. While this proved to be a sound approach, an alternative consideration was semi-supervised machine learning, which leverages existing labels while also identifying new patterns.

Zooming in on the selection of unsupervised machine learning, four clustering algorithms were considered, and two were tested. Ultimately, the decision was made to proceed with K-Means. In retrospect, the initial choice of K-Means, based on its comprehensibility and ease of implementation, is validated. However, a deeper exploration of DBSCAN reveals a potential standard method for determining Eps and MinPts, making it a valuable alternative. Due to time constraints, the decision was made to focus on K-Means, an algorithm that demonstrated effectiveness and ease of use from the outset. In the application of both K-Means and DBSCAN, various decisions were required throughout the algorithmic steps, such as vectorization and dimension reduction. The TF-IDF vectorizer emerged as the most effective for this research, capturing word order and frequency. While dimension reduction was set at two for simplicity and computational efficiency, exploring other dimensions could enhance results, although this aspect was not covered in this research.

A substantial portion of this thesis investigates the structuring of input data for the algorithm. Achieving uniformity in the imported data is crucial for optimal clustering within the chosen algorithm. Different data preparation tools were tested for effectiveness and compatibility. Notably, the order in which these tools are applied influences the end result. Although various combinations were not exhaustively tested due to the use of six different tools, there is potential for further optimization in determining the optimal order of these tools.In this exploration on the used data, it became evident that not all data preparation tools reached their full potential, particularly when faced with the presentation of data in within the project planning, since data inconsistency can be found. This inconsistency stems from multiple contributor. Notably, challenges arise from the use of two languages, the inclusion of abbreviations, and instances of spelling mistakes.

The first challenge relates to the use of two languages within the project. While the base planning predominantly used English for activity descriptions, some activities were documented in Dutch. Next to this the projects used for validation were completely in Dutch. This linguistic inconsistency called for a translation process. During translation, the chosen translation tool might opt for synonyms not originally used by planners in the respective language. Consequently, the clustering algorithm, relying on text vectorization, perceives these translated words as distinct, leading to their placement in separate clusters. Therefore, a more optimal presentation of the data would involve maintaining uniformity in language usage. Abbreviations pose another hurdle in the data. Their usage becomes problematic when scaling up the dataset, especially when project-specific abbreviations are employed. As these abbreviations may not be applicable to other projects, finding similarities becomes challenging. Deciding on the abbreviations to keep and eliminate requires finding a balance, as the complete removal of abbreviations may result in the loss of essential content within activity names. Expanding abbreviations to their full form can simplify this process, allowing the removal of project-

specific terms while preserving the core content in the activity name. Finally, addressing spelling mistakes is crucial in mitigating data inconsistency. Difference in the spelling of words can lead to varying interpretations by the algorithm, causing the same word, spelled differently, to be assigned to different clusters. Rectifying these spelling errors is needed to enhancing the accuracy of the clustering algorithm's outcome.

Moving forward in the process involves assigning names to the clusters. This naming was determined based on the most frequently occurring activity within each cluster. However, upon verification, a notable number of clusters exhibited low coherence. This gives rise to several issues. Firstly, however not the most immediate concern, the naming of clusters may inadequately capture the cluster's content since it relies solely on the most frequent activity. Given that this represents only a fraction of the cluster, it fails to capture the complete essence. Furthermore, the low internal similarity within the cluster points to a more significant problem, the inadequacy of the clustering process itself. The suggestions discussed already this chapter could address these issues and enhance the clustering performance. During the evaluation of clustering, manual adjustments can be implemented to refine the results further. This involves a in depth examination of clusters and potentially combining or reorganizing them. While effective, this manual approach is not the most optimal solution in regards to digitalization. Hence, a more beneficial approach for refining the clustering, could involve exploring semi-supervised learning. Using the outcomes of K-means as a foundation for correctly clustered activities, semi-supervised learning could delve deeper into the data, uncovering links and patterns to enhance overall clustering accuracy.

Despite the suboptimal level of similarity within the clusters, the testing of labelling for new planning activities within the established clusters was carried out. This was undertaken to gain a deeper understanding of the labelling process and to extract valuable insights from the validation process. Notably, during this labelling process, it became apparent that clusters with already low similarity were predominantly utilized as labels for new activities. Additionally, labelling predominantly relied on the verbs within activity names, given their higher coherence across the base planning used for clustering and the activities in the new planning. Moreover, the earlier discussed translation process did not contribute positively to the labelling process. To enhance labelling accuracy, it is imperative to improve cluster similarity. Activities should align in format with the base file, and optimal conditions for this would involve both project plannings being in the same language, undergoing identical data preparation, and employing consistent terminology.

In summary, this study delves into the clustering and labelling of project planning activities, highlighting the complexities and improvements in the data-driven methods applied. The decisions made, ranging from algorithm selection to data preparation and cluster naming, reflect a practical balance between convenient and optimal outcomes. While unsupervised machine learning, particularly the K-Means algorithm, proved a convenient use, the potential of alternative methods, like DBSCAN, suggests options for further exploration. The discussion on data preparation tools emphasizes the importance of data uniformity and refining the order of application. Addressing language inconsistencies, abbreviations, and spelling mistakes emerged as crucial steps in improving the clustering and labelling processes. Manual adjustments and exploring semi-supervised learning offer promising paths for refining clustering results and overcoming digital limitations. The focus on improving cluster similarity is crucial for accurate cluster naming and meaningful labelling of new planning activities.

7. Conclusion

7.1. Main contributions

This thesis explores automation improvement within the AEC industry, focusing on planning and scheduling. The main objectives presented in this research are to develop a mechanism for clustering and labelling planning activities in construction projects to enhance the evaluation between the planning of different projects. This mechanism will facilitate comparing planning activities, allowing project managers and planners to quickly identify patterns and potential issues in the planning process, ultimately leading to more efficient project management and improved project outcomes. Such a systematic method for evaluation purposes is useful for pinpointing potential delays and making realistic assumptions on activity duration while making a new plan. However, the industry's reliance on manual processes has hindered planning efficiency in complex construction projects. In this thesis, the aim was supported by main and sub-research questions. The main research question is:

"How can machine learning contribute to the effective evaluation of project planning across different construction projects?"

The sub-questions supporting this main question are:

1. “What is the current state-of-the-art knowledge of planning principles, artificial intelligence and the implementation of Artificial Intelligence into planning?”
2. “How can machine learning techniques cluster planning activities based on task similarity?”
3. “How can planning activities of the same type be successfully clustered together?”
4. “What are the essential criteria and requirements for accurately labelling new planning activities and assigning them to clusters with similar activities?”
5. “To what extent does implementing the clustering and labelling mechanism work efficiently over multiple projects?”

These questions formed the basis for exploring digitalization in project planning evaluation, focusing on the evaluation between the planning of different projects. To fulfil the aim, the research was divided into three objectives. First, the clustering process was developed to identify the different types of planning activities based on their names. The first three sub-research questions correspond to this objective. Second, the development of the labelling process. This process concentrates on assigning new project activities to the appropriate categories made in the clustering process. This objective is in line with the fourth sub-research question. Third, the effectiveness of this clustering and labelling will be evaluated. This last objective correlates to the last sub-research question. This conclusion will evaluate the three objectives and their belonging research questions.

The clustering process will be evaluated within the first objective. To do this correctly, first, the state-of-the-art is covered. The first sub-research question guides the research on determining what part of the project planning process can be digitalized. This research involved a systematic literature review, revealing existing research in the digitalization field, particularly in response to emerging trends such as AI and automation. Notably, efforts have been made to integrate these technologies within the AEC sector. However, a predominant focus on completely automating the project planning process has led to the generation of numerous assumptions during the transition from manual to automated planning. These assumptions pose a challenge, hindering the seamless application of developed mechanisms and methodologies to diverse projects. Consequently, the difficulty arises in scaling up these methodologies for application to other projects where similar assumptions may not hold true. Therefore, it was decided that this research would focus on getting a semi-automated approach. It

would not focus on automated planning but on how automatization could help the planning evaluation process. This research evaluates different projects by categorising the same kind of activities. Furthermore, the literature chapter also dives into the automatization aspects in general. This means that the meaning and development of artificial intelligence and its future possibilities are covered. Forming the basis of the rest of the thesis, in which these artificial intelligence and machine learning aspects were used. Recognizing the limitations of this approach, the decision was made to concentrate on achieving a semi-automated methodology. This shift entails a focus not on fully automated planning but on leveraging automation to enhance the evaluation process of a planning. Specifically, the research centres on evaluating different plans by categorizing similar activities. This literature chapter serves as the foundation for developing the algorithm using AI, ML, and NLP.

Subsequently, these findings are used to introduce an approach for clustering planning activities. The exploration of AI and ML formed the basis for the methodology of this thesis, subsequently answering the second sub-research question. The methodology zoomed in further on the clustering aspects of ML, an aspect of unsupervised machine learning that was touched upon in the literature chapter. The different types of unsupervised clustering are elaborated on, and with the help of an MCA, the suitability of the different clusters is scored. This concluded with DBSCAN and K-means scoring as the most suitable options. After a test case with both algorithms, the decision was made to continue the research with K-means. K-means worked best because of the ease of use and the limited variables. Only the amount of K (clusters) needed to be defined beforehand. Since the amount of K could be optimized using the Silhouette score, it was ensured that the most suitable amount of K was defined between the set boundaries.

When using K-means on the entire dataset, some computational issues were found. The cause was the extensive amount of data that was not always in the same format or style. Therefore, different types of data preparation were done to create a uniform method eventually. These preparations include: the use of only nouns and verbs, the removal of punctuation marks and text between brackets, lemmatizing the words, translating the data to English, and the removal of project-specific terms. After testing different data preparation methods, the method that used all the suggested preparation tools and lemmatization has been proven to work the best. This decision for data preparation also translates to the effectiveness of the clustering. The final clustering gave a Silhouette score of 0.596, the highest silhouette score out of all possible data preparation options and was also chosen. Since this score is generally perceived as okay, it still means that every cluster does not have a uniformly high similarity. Naturally, this leads to certain drawbacks. This score resulted in some clusters with a low similarity, which can be seen as a kind of “Dump” cluster where all the remaining activities that did not fit in another cluster were placed. To fix this, the Silhouette score needs to be increased. This can be done in several ways: manually improving the results, improving the data with data preparation, and changing the clustering technique. The data preparation part to raise this score is researched extensively in this research. Therefore, this research advises that to raise this score, another clustering technique is used.

The manual approach also has been researched in this thesis. To conclude, the manual approach was found to be not feasible for this research. This changed the number of clusters, making the K-means clustering file not work to its full potential. Therefore, the decision is not to go further with these manual alterations but to keep the automated generated clusters.

In general, it can be said that K-Means proves its effectiveness in the small data sample. However, when upscaling the data, the complexity increases, resulting in the clustering, which proved less uniform. What can be seen during the validation process is that the new projects that employ the labelling algorithm do not work as intended. The validation process indicates that the clusters with low similarity

within a cluster are most often used. Within the unsupervised clustering, this will most likely always be the problem. Therefore, searching for another method to reach the research goal is suggested. What is most important is that an algorithm can learn from multiple iterations of clustering and labelling results. It is necessary to teach an algorithm its strong and weak points while clustering to achieve the desired results.

7.2. Implications for industry and research

As mentioned in the introduction, the AEC industry is dealing with the industry's slow adoption of digital technologies in project management. This thesis tried to address this issue and focus on the digitalization of a part of the project management process, namely the planning. This section will highlight the contribution of this research to the industry itself, as well as the research field.

First, the industry implications of the provided approach and mechanism will be explained. The research addresses the industry's challenge of manual processes hindering planning efficiency by offering a systematic approach to categorising and comparing planning activities. This research aims to let project managers and planners quickly identify patterns and potential issues in project planning, ultimately enhancing project management efficiency and improving project outcomes. The results generated in this research are not advanced and detailed enough to identify these patterns. However, there are still other implications for industry regarding this research. First, the approach researched focused on minimizing the incorporation of manual work. By employing the unsupervised ML, no manual labels need to be made. This path has been taken to avoid time-consuming tasks when this project would have been used within the industry. However, as experienced during this research, the approach did not generate the desired results. One can argue that only employing unsupervised ML will not generate the desired outcome. This outcome can be seen as a valuable insight. Hence, if this approach would have worked, it would have been a less time-consuming and complex method for industry users.

In the realm of research, this thesis explores the digitalization of project planning evaluation. By addressing the main research question and its sub-questions, the research contributes to the broader understanding of how machine learning can effectively evaluate project planning across different construction projects. Focusing on a semi-automated approach rather than fully automated planning is highlighted as a strategic choice. As explored in this research, this approach proves to be less time-consuming and more feasible, making it an essential first step in exploring automation improvement within the AEC industry. Researchers in the field can use this approach as a benchmark or starting point for more advanced studies in the future. Regarding the further implications of the digitalization of the industry, Research can shift towards incorporating ML and AI within project planning evaluations while still being accessible and understandable for its users within the industry.

7.3. Future work

Significant decisions were made throughout the research to shape the methodology and outcomes. Reflecting on these choices opens up possibilities for further exploration and refinement of the proposed approach. Future work can build on the foundations established in this thesis and delve into several key areas.

First, the choice of the unsupervised machine learning algorithm, particularly the emphasis on K-Means clustering, has been a significant decision within this research. However, the exploration of ML and AI can be broadened for future work. Semi-supervised or Self-supervised ML can improve aspects in which unsupervised ML comes short. Incorporating elements of supervised learning within the broader unsupervised framework could leverage the benefits of both approaches. Semi-supervised learning allows the algorithm to learn from labelled and unlabelled data. In the context of project planning,

where certain activities may be known and labelled while others are not, a semi-supervised approach could enhance the accuracy of clustering and labelling. Future research can explore integrating project-specific knowledge or historical data to guide the algorithm, potentially resulting in more refined and contextually relevant outcomes. Self-supervised learning, conversely, involves training a model using its own generated labels from the input data, eliminating the need for external labelling. In the context of project planning evaluation, where obtaining labelled data may be challenging or resource-intensive, self-supervised learning provides a promising alternative. Comparative studies in the AEC industry across supervised, unsupervised, semi-supervised, and self-supervised paradigms can provide an in-depth understanding of their strengths and weaknesses. Based on the specific characteristics and challenges of project planning datasets, a study like this could inform researchers and industry professionals about the best machine learning approach.

Second, while the current research has concentrated on clustering planning activities based on their names, the dataset can be enriched to incorporate additional contextual information. Including project segment, Work Breakdown Structure (WBS) configuration, or temporal duration can contribute to a more comprehensive understanding of planning activities. This dataset expansion allows for a more nuanced clustering analysis, providing a refined perspective on the relationships between different project elements. Including temporal aspects, for instance, could reveal patterns related to project timelines and scheduling, offering valuable insights into potential bottlenecks or areas of optimization. Similarly, considering project segments and WBS configurations can provide a deeper understanding of how different project parts contribute to the overall planning landscape. Future research can explore the impact of such enriched datasets on the clustering and labelling mechanisms, thereby enhancing the overall effectiveness of the evaluation process.

Third, an important aspect of future work involves the development of iterative learning mechanisms within the algorithms employed. The research suggests that an algorithm capable of learning from multiple iterations of clustering and labelling results can better adapt to the complexities of project planning. Future research can explore the implementation of feedback loops that allow the algorithm to refine its understanding over time, continually improving its ability to categorize and cluster planning activities accurately. Moreover, the investigation into manual interventions to improve clustering outcomes highlights the potential for a hybrid approach. Future work can explore the integration of human expertise to guide the algorithm's learning process, combining the strengths of automation with human insights to achieve more robust and contextually relevant results.

In conclusion, future research in this domain can benefit from a more extensive exploration of algorithms, an enriched dataset, and the development of adaptive learning mechanisms to ensure the continual improvement of the proposed approach. These advancements would contribute to a more sophisticated and adaptable system for evaluating project planning within the AEC industry.

References

Amasha, M. A., Khairy, D., Abougalala, R. A., Alkhalaf, S., Amashaa, M. A., Khairya, D., Abougalalaa, R. A., & Areed, M. F. (2020). Python-based Graphical User Interface for Automatic Selection of Data Clustering Algorithm Natural and Technical sciences. *International Journal of Future Generation Communication and Networking*, *13*(3), 451–461. https://www.researchgate.net/publication/341741906

Antonijevic, M., Zivkovic, M., Arsic, S., & Jevremovic, A. (2020). Using AI-Based Classification Techniques to Process EEG Data Collected during the Visual Short-Term Memory Assessment. *Journal of Sensors*, *2020*, 1–12. https://doi.org/10.1155/2020/8767865

Arya, S. (2018). Parentheses. *Canadian Medical Association Journal*, *190*(3), E82–E83. https://doi.org/10.1503/cmaj.171031

Balakrishnan, V., & Ethel, L.-Y. (2014). Stemming and Lemmatization: A Comparison of Retrieval Performances. *Lecture Notes on Software Engineering*, *2*(3), 262–267. https://doi.org/10.7763/LNSE.2014.V2.134

Boland, A., Cherry, M. G., & Dickson, R. (2017). Doing a Systematic Review. *Sage Publishing*.

Bouyssou, D., & Marchant, T. (2011). Bibliometric rankings of journals based on Impact Factors: An axiomatic approach. *Journal of Informetrics*, *5*(1), 75–86. https://doi.org/10.1016/j.joi.2010.09.001

Bridges, C. C. (1966). Hierarchical Cluster Analysis. *Psychological Reports*, *18*(3), 851–854. https://doi.org/10.2466/pr0.1966.18.3.851

Brotherton, S. A. , F. R. T. , & N. E. S. (2008). *Applying the work breakdown structure to the project management lifecycle*.

Chaudhuri, S., Ganti, V., & Kaushik, R. (2006). *A Primitive Operator for Similarity Joins in Data Cleaning*.

Corrales, D., Corrales, J., & Ledezma, A. (2018). How to Address the Data Quality Issues in Regression Models: A Guided Process for Data Cleaning. *Symmetry*, *10*(4), 99. https://doi.org/10.3390/sym10040099

Dalmaijer, E. S., Nord, C. L., & Astle, D. E. (2022). Statistical power for cluster analysis. *BMC Bioinformatics*, *23*(1). https://doi.org/10.1186/s12859-022-04675-1

Deng, S., Tian, Y., & Zhang, H. (2009). Using the Bibliometric Analysis to Evaluate Global Scientific Production of Data Mining Papers. *2009 First International Workshop on Database Technology and Applications*, 233–238. https://doi.org/10.1109/DBTA.2009.157

Derakhshanfar, H., Ochoa, J. J., Kirytopoulos, K., Mayer, W., & Tam, V. W. Y. (2019). Construction delay risk taxonomy, associations and regional contexts. *Engineering, Construction and Architectural Management*, *26*(10), 2364–2388. https://doi.org/10.1108/ECAM-07-2018-0307

Eck, N. J. van, & Waltman, L. (2023). *VOSviewer*. https://www.vosviewer.com/contact

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*. www.aaai.org

Fahim, A. (2022). An Extended DBSCAN Clustering Algorithm. In *IJACSA) International Journal of Advanced Computer Science and Applications* (Vol. 13, Issue 3). www.ijacsa.thesai.org

Fuchs, M., & Höpken, W. (2022). *Clustering* (pp. 129–149). https://doi.org/10.1007/978-3-030-88389-8_8

Fukunaga, K., & Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, *21*(1), 32–40. https://doi.org/10.1109/TIT.1975.1055330

Hajdu, M. (2018). Survey of precedence relationships: Classification and algorithms. *Automation in Construction*, *95*, 245–259. https://doi.org/10.1016/j.autcon.2018.08.012

Hassan, F. U., Le, T., & Le, C. (2023). Automated Approach for Digitalizing Scope of Work Requirements to Support Contract Management. *Journal of Construction Engineering and Management*, *149*(4). https://doi.org/10.1061/JCEMD4.COENG-12528

Hassler, M., & Fliedl, G. (2006). Text preparation through extended tokenization. *Data Mining VII: Data, Text and Web Mining and Their Business Applications*, 13–21. https://doi.org/10.2495/DATA060021

Hintze, A. (2017). Data Preprocessing. In *Encyclopedia of Machine Learning and Data Mining* (pp. 327–327). Springer US. https://doi.org/10.1007/978-1-4899-7687-1_100100

Hossain, Md. Z., Islam, Md. J., Miah, Md. W. R., Rony, J. H., & Begum, M. (2021). Develop a dynamic DBSCAN algorithm for solving initial parameter selection problem of the DBSCAN algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*, *23*(3), 1602. https://doi.org/10.11591/ijeecs.v23.i3.pp1602-1610

Hosseini, M. R., & Chileshe, N. (2014). *Emerging ICT-Based Methods in the Architecture, Engineering, and Construction Context* (pp. 7086–7095). https://doi.org/10.4018/978-1-4666-5888-2.ch698

Howard, H. C., Levitt, R. E., Paulson, B. C., Pohl, J. G., & Tatum, C. B. (1989). Computer Integration: Reducing Fragmentation in AEC Industry. *Journal of Computing in Civil Engineering*, *3*(1), 18–32. https://doi.org/10.1061/(ASCE)0887-3801(1989)3:1(18)

Jeatrakul, P., Wong, K. W., & Fung, C. C. (2010). Data Cleaning for Classification Using Misclassification Analysis. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, *14*(3), 297–302. https://doi.org/10.20965/jaciii.2010.p0297

Jeena, S., Chaudhary, A., & Thakur, A. (2023). Implementation & Analysis of Online Retail Dataset Using Clustering Algorithms. *2023 4th International Conference on Intelligent Engineering and Management (ICIEM)*, 1–6. https://doi.org/10.1109/ICIEM59379.2023.10166552

Jordan, J., & Angelopoulou, E. (2013). Mean-shift clustering for interactive multispectral image analysis. *2013 IEEE International Conference on Image Processing*, 3790–3794. https://doi.org/10.1109/ICIP.2013.6738781

José de Oliveira, O., Francisco da Silva, F., Juliani, F., César Ferreira Motta Barbosa, L., & Vieira Nunhes, T. (2019). Bibliometric Method for Mapping the State-of-the-Art and Identifying Research Gaps and Trends in Literature: An Essential Instrument to Support the

Development of Scientific Projects. In *Scientometrics Recent Advances*. IntechOpen. https://doi.org/10.5772/intechopen.85856

Kitasuka, T., Aritsugi, M., & Rahutomo, F. (2012). *Semantic Cosine Similarity*. https://www.researchgate.net/publication/262525676

Kulkarni, A., & Shivananda, A. (2021). Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning Using Python. In *Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning Using Python*. Apress Media LLC. https://doi.org/10.1007/978-1-4842-7351-7

Lamba, S., Saini, P., Kukreja, V., & Sharma, B. (2021). Role of Mathematics in Machine Learning. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3833931

Li, D., & Lu, M. (2019). Classical Planning Model-Based Approach to Automating Construction Planning on Earthwork Projects. *Computer-Aided Civil and Infrastructure Engineering*, *34*(4), 299–315. https://doi.org/10.1111/mice.12420

Luchi, D., Loureiros Rodrigues, A., & Miguel Varejão, F. (2019). Sampling approaches for applying DBSCAN to large datasets. *Pattern Recognition Letters*, *117*, 90–96. https://doi.org/10.1016/j.patrec.2018.12.010

MacQueen, J., Le Cam, L. M., & Neyman, J. (1967). SOME METHODS FOR CLASSIFICATION AND ANALYSIS OF MULTIVARIATE OBSERVATIONS. *Proc. 5thBerkeley Symp. Math. Statistics AndProbability.*, *1*, 281–297.

Mahdavian, A., & Shojaei, A. (2020). Hybrid Genetic Algorithm and Constraint-Based Simulation Framework for Building Construction Project Planning and Control. *Journal of Construction Engineering and Management*, *146*(12). https://doi.org/10.1061/(ASCE)CO.1943-7862.0001939

Mahesh Kumar, K., & Rama Mohan Reddy, A. (2016). A fast DBSCAN clustering algorithm by accelerating neighbor searching using Groups method. *Pattern Recognition*, *58*, 39–48. https://doi.org/10.1016/j.patcog.2016.03.008

Manzoor, B., Othman, I., & Pomares, J. C. (2021). Digital technologies in the architecture, engineering and construction (Aec) industry—a bibliometric—qualitative literature review of research activities. *International Journal of Environmental Research and Public Health*, *18*(11). https://doi.org/10.3390/ijerph18116135

Marcus, G. (2020). *The Next Decade in AI: Four Steps Towards Robust Artificial Intelligence*.

Marsani, A., & Acharya, A. (n.d.). *Evaluation of Pedestrian LOS for Footpath by Cluster Analysis and Questionnaire Survey: A case Study of Kathmandu*. https://www.researchgate.net/publication/359789369

Mbala, M., Aigbavboa, C., & Aliu, J. (2019). Causes of delay in various construction projects: A literature review. *Advances in Intelligent Systems and Computing*, *788*, 489–495. https://doi.org/10.1007/978-3-319-94199-8_47

McKinsey & Company. (2020). *The next normal in construction*.

Michałowska-Kaczmarczyk, A. M., & Michałowski, T. (2014). Compact Formulation of Redox Systems According to GATES/GEB Principles. *Journal of Analytical Sciences, Methods and Instrumentation*, *04*(02), 39–45. https://doi.org/10.4236/jasmi.2014.42006

Murtagh, F., & Legendre, P. (2014). Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*, *31*(3), 274–295. https://doi.org/10.1007/s00357-014-9161-z

Murugesan, N., Cho, I., & Tortora, C. (2021). *Benchmarking in Cluster Analysis: A Study on Spectral Clustering, DBSCAN, and K-Means* (pp. 175–185). https://doi.org/10.1007/978-3-030-60104-1_20

Mythili, M. S., & Mohamed Shanavas, A. R. (2013). Performance Evaluation of Apriori and FP-Growth Algorithms. *International Journal of Computer Applications*, *79*(10), 34–37. https://doi.org/10.5120/13779-1650

Nagarajappa, R. K. B., & Suresh, Y. (2023). Software Development Effort Estimation Using Relational Database and Optimized Learning Mechanism. *Journal of Computer Science*, *19*(4), 540–553. https://doi.org/10.3844/jcssp.2023.540.553

Naik Gaonkar, M., & Sawant, K. (2013). *AutoEpsDBSCAN : DBSCAN with Eps Automatic for Large Dataset* (Issue 2).

Nielsen, F. (2016). *Hierarchical Clustering* (pp. 195–211). https://doi.org/10.1007/978-3-319-21903-5_8

O'Brien, J. J., & Plotnick, F. L. (2005). *CPM in construction management.* McGraw-Hill Professional.

O'Connor, J. (2001). *The NLP workbook : a practical guide to achieving the results you want* (2013th ed.). Conari Press.

Ogbuabor, G., & F. N, U. (2018). Clustering Algorithm for a Healthcare Dataset Using Silhouette Score Value. *International Journal of Computer Science and Information Technology*, *10*(2), 27–37. https://doi.org/10.5121/ijcsit.2018.10203

Oladipupo, T. (2010). Types of Machine Learning Algorithms. In *New Advances in Machine Learning*. InTech. https://doi.org/10.5772/9385

Ouchchy, L., Coin, A., & Dubljević, V. (2020). AI in the headlines: the portrayal of the ethical issues of artificial intelligence in the media. *AI and Society*, *35*(4), 927–936. https://doi.org/10.1007/s00146-020-00965-5

Ozge Ozkok, F. (2017). *International Journal of Intelligent Systems and Applications in Engineering A New Approach to Determine Eps Parameter of DBSCAN Algorithm*. www.atscience.org/IJISAE

Pandey, P. (2020, October 22). *DBSCAN Clustering*. Https://Machinelearninggeek.Com/Dbscan-Clustering/.

Panesar, A. (2021). What Is Artificial Intelligence? In *Machine Learning and AI for Healthcare* (pp. 1–18). Apress. https://doi.org/10.1007/978-1-4842-6537-6_1

Pellerin, R., & Perrier, N. (2019). A review of methods, techniques and tools for project planning and control. *International Journal of Production Research*, *57*(7), 2160–2178. https://doi.org/10.1080/00207543.2018.1524168

Pérez-Ortega, J., Nely Almanza-Ortega, N., Vega-Villalobos, A., Pazos-Rangel, R., Zavala-Díaz, C., & Martínez-Rebollar, A. (2020). The K-Means Algorithm Evolution. In *Introduction to Data Science and Machine Learning*. IntechOpen. https://doi.org/10.5772/intechopen.85447

Perkins, J. (2014). *Python 3 text processing with NLTK 3 cookbook : over 80 practical recipes on natural language processing techniques using Python's NLTK 3.0* (F. Fattohi, Ed.; 2nd ed.). Packt Publishing Ltd.

PRISMA. (n.d.). *PRISMA*. Http://Prisma-Statement.Org/. Retrieved April 14, 2023, from http://prisma-statement.org/

Rahman, S., & Kandogan, E. (2022, April 29). Characterizing Practices, Limitations, and Opportunities Related to Text Information Extraction Workflows: A Human-in-the-loop Perspective. *Conference on Human Factors in Computing Systems - Proceedings*. https://doi.org/10.1145/3491102.3502068

Relatics. (2023). *Relatics*. Https://Www.Relatics.Com/. https://www.relatics.com/

Ren, Y., Kamath, U., Domeniconi, C., & Zhang, G. (2014). *Boosted Mean Shift Clustering* (pp. 646–661). https://doi.org/10.1007/978-3-662-44851-9_41

Rianto, Mutiara, A. B., Wibowo, E. P., & Santosa, P. I. (2021). Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation. *Journal of Big Data*, *8*(1), 26. https://doi.org/10.1186/s40537-021-00413-1

Rizqi Lahitani, A., Erna Permanasari, A., & Akhmad Setiawan, N. (n.d.). *Cosine Similarity to Determine Similarity Measure: Study Case in Online Essay Assessment*.

Roberts, C. J., Pärn, E. A., Edwards, D. J., & Aigbavboa, C. (2018). Digitalising asset management: concomitant benefits and persistent challenges. *International Journal of Building Pathology and Adaptation*, *36*(2), 152–173. https://doi.org/10.1108/IJBPA-09-2017-0036

Rodríguez, P., Bautista, M. A., Gonzàlez, J., & Escalera, S. (2018). Beyond one-hot encoding: Lower dimensional target embedding. *Image and Vision Computing*, *75*, 21–31. https://doi.org/10.1016/j.imavis.2018.04.004

Roelleke, T., & Wang, J. (2008). TF-IDF uncovered. *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 435–442. https://doi.org/10.1145/1390334.1390409

Roman, D. D. (1962). The PERT System: An Appraisal of Program Evaluation Review Technique. *Academy of Management Journal*, *5*(1), 57–65. https://doi.org/10.2307/254602

Rozenes, S., Vitner, G., & Spraggett, S. (2006). Project Control: Literature Review. *Project Management Journal*, *37*(4), 5–14. https://doi.org/10.1177/875697280603700402

Sacks, R., Eastman, C., Lee, G., & Teicholz, P. (2018). *BIM Handbook*. John Wiley & Sons, Inc. https://doi.org/10.1002/9781119287568

Sagita, L., Supriadi, R., Latief, Y., Susilo, B., & Rajasa, M. (2017). DEVELOPMENT OF RISK-BASED STANDARDIZED WBS (WORK BREAKDOWN STRUCTURE) FOR COST ESTIMATION OF APARTMENT'S PROJECT. In *International Journal of Civil Engineering and Technology (IJCIET)* (Vol. 8, Issue 10).

http://http://iaeme.comhttp://iaeme.com/Home/journal/IJCIET823editor@iaeme.comhttp://iaeme.com

Samir, A., & Lahbib, Z. (2018). *Stemming and Lemmatization for Information Retrieval Systems in Amazigh Language* (pp. 222–233). https://doi.org/10.1007/978-3-319-96292-4_18

Sander, J., Ester, M., Kriegel, H.-P., & Xu, X. (1998). Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Mining and Knowledge Discovery*, *2*(2), 169–194. https://doi.org/10.1023/A:1009745219419

Saravanan, R., & Sujatha, P. (2018). A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification. *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, 945–949. https://doi.org/10.1109/ICCONS.2018.8663155

Sawhney, A., Riley, M., & Irizarry, J. (2020). *Construction 4.0* (A. Sawhney, M. Riley, & J. Irizarry, Eds.). Routledge. https://doi.org/10.1201/9780429398100

Seger, C. (2018). An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing. In *DEGREE PROJECT TECHNOLOGY*.

Sippel, K., Kübler, T., Fuhl, W., Schievelbein, G., Rosenberg, R., & Rosenstiel, W. (2015). Eyetrace2014 - Eyetracking Data Analysis Tool. *Proceedings of the International Conference on Health Informatics*, 212–219. https://doi.org/10.5220/0005352902120219

Surjandari, I., Dhini, A., Wibisana, N., & Lumbantobing, E. W. I. (2015). University Research Theme Mapping: A Co-word Analysis of Scientific Publications. *International Journal of Technology*, *6*(3), 410. https://doi.org/10.14716/ijtech.v6i3.1462

The Oxford Dictionary of Phrase and Fable. (2005). Oxford University Press. https://doi.org/10.1093/acref/9780198609810.001.0001

Torkanfar, N., & Rezazadeh Azar, E. (2020). Quantitative similarity assessment of construction projects using WBS-based metrics. *Advanced Engineering Informatics*, *46*. https://doi.org/10.1016/j.aei.2020.101179

Tranfield, D., Denyer, D., & Smart, P. (2003). Towards a Methodology for Developing Evidence-Informed Management Knowledge by Means of Systematic Review. *British Journal of Management*, *14*(3), 207–222. https://doi.org/10.1111/1467-8551.00375

Wen, S., Tang, H., Ying, F., & Wu, G. (2023). Exploring the Global Research Trends of Supply Chain Management of Construction Projects Based on a Bibliometric Analysis: Current Status and Future Prospects. *Buildings*, *13*(2). https://doi.org/10.3390/buildings13020373

Werbrouck, J., Pauwels, P., & Beetz, J. (2019). *Towards a Decentralised Common Data Environment using Linked Building Data and the Solid Ecosystem*. https://www.w3.org/standards/semanticweb/ontology.html,

Wu, C., Chen, D., & Huang, S. (2018). Complexity of Construction Mega Infrastructure Project. *Complexity*, *2018*, 1–1. https://doi.org/10.1155/2018/7382083

Yang, K., & Lee, J. (2013). Bibliometric Approach to Research Assessment: Publication Count, Citation Count, & Author Rank. *Journal of Information Science Theory and Practice*, *1*(1), 27–41. https://doi.org/10.1633/JISTaP.2013.1.1.2

Zecheru, V., & Olaru, B. G. (2016). Work Breakdown Structure (WBS) in Project Management. In *Review of International Comparative Management* (Vol. 17, Issue 1).

Zupic, I., & Čater, T. (2015). Bibliometric Methods in Management and Organization. *Organizational Research Methods*, *18*(3), 429–472. https://doi.org/10.1177/1094428114562629

Appendix A: Complete bibliometric literature review

In the Systematic literature review, four main themes were explored in that research: Project Planning, Project Control, Artificial Intelligence, and Classification Techniques. Project Planning and Artificial Intelligence are the main areas of interest. The aim is to explore these themes quantitatively by analyzing articles focused on them, resulting in two separate maps: one for Project Planning and another for Artificial Intelligence. These two maps will be generated as followed; one encompassing all the articles from the search query in Equation 1 that include the keyword Project Planning, and another focusing on articles containing the keyword Artificial Intelligence. For both of these filtered search queries an CSV output is made of all relevant articles. An important aspect to note is that this bibliometric CSV output should include the information that will be researched, which is in this case the article keywords. For the analysis of these files VOSviewer will be used. This is a software program specifically designed for bibliographic analyses (Eck & Waltman, 2023). In VOSviewer, various options for map creation are available, including network data, bibliographic, or text data. In order to meet the requirements of this research and focus on the co-occurrence of keywords within the bibliometric data, the bibliographic data option is chosen. By examining the keyword co-occurrence, important keywords and clusters of keywords can be identified, leading to a clearer overview of the research field's saturation.

Project Planning

The bibliometric map, shown in Figure 26, is based on the articles focusing on Project Planning. Within this map six distinct clusters can be found. In general the words that have the most occurrences are the words, Project Planning, Artificial Intelligence, and Project Management. These keywords with its occurrences and all other keywords and occurrences can be found in Table 11. Apart from the six clusters some other specifics of the map can be named. The total number of items, these are the analysed keywords (in the rest of this paper the word keywords will be used instead of items), are 29, the total links are 263, and lastly the total link strength is 135.50. These numbers result in an average link strength per link of 0.52 and on average 9 links per keyword.

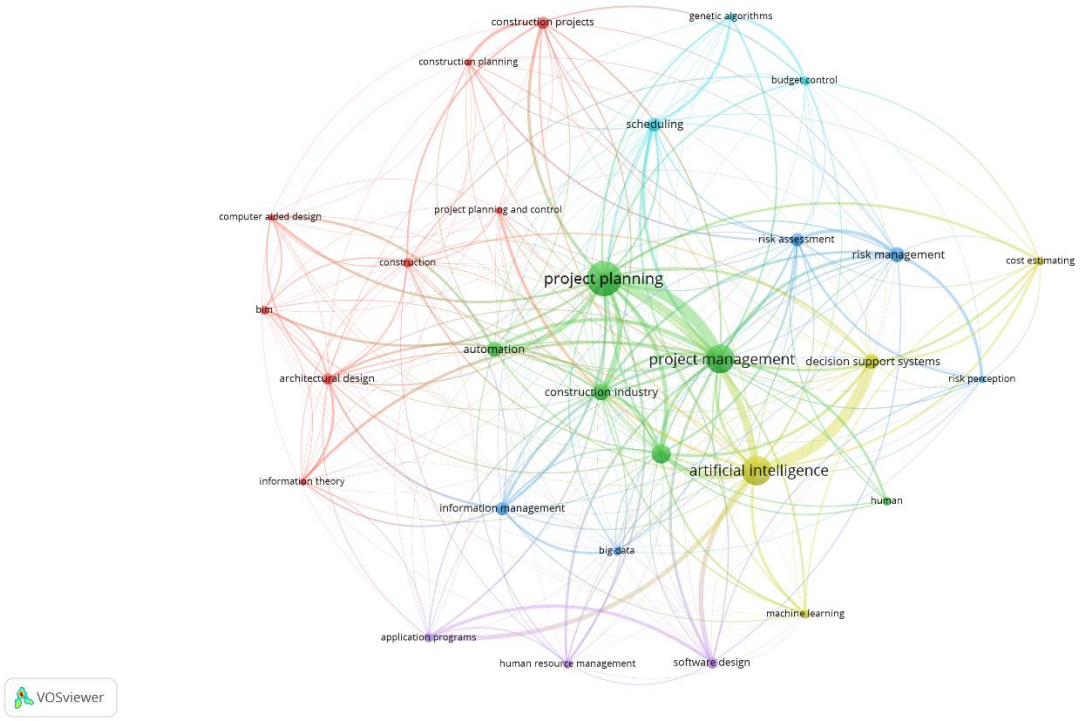


Figure 26 Bibliometric map of Keywords of Project Planning articles based on occurrences

The way this bibliometric map is analysed is first looking at the different clusters present in the map, in this case six. These six clusters will be discussed one by one, to show the different themes of the clusters and the way they are structured. After looking at each individual cluster, the overall map is analysed. This will focus more on the link strength between keywords, but also the connection of the clusters between each other.

Table 11 Project Planning keywords map with occurrence per keyword

Cluster 1 *Red*		**Cluster 2** *Green*		**Cluster 3** *Blue*		**Cluster 4** *Yellow*		**Cluster 5** *Purple*		**Cluster 6** *Aqua*	
Construction Projects	9	Project Planning	33	Risk Management	12	Artificial Intelligence	27	Software Design	8	Scheduling	11
Architectural Design	8	Project Management	26	Information Management	10	Decision Support Systems	12	Application Programs	7	Budget Control	6
Construction	7	Decision Making	16	Risk Assessment	10	Cost Estimating	6	Human Resource Management	6	Genetic Algorithms	5
BIM	6	Construction Industry	13	Big Data	6	Machine Learning	6				
Computer Aided Design	5	Automation	12	Risk Perception	5						
Construction Planning	5	Human	6								
Information Theory	5										
Project Planning and Control	5										

Cluster 1 (Red): This is the largest cluster based on the number of items. However, based on occurrence this is not the case. In this cluster “construction projects” is the largest item, with 9 occurrences, followed by Architectural Design and Construction. The keywords in this cluster suggest a focus on various aspects related to architectural design, construction projects, and project planning and control. The highest occurrence in this cluster for "Construction Projects" implies that this item has the most significant research interest in this specific area within this cluster. Furthermore, the occurrence of "Architectural Design," "BIM," and "Computer Aided Design" indicates the importance of technological advancements and digital tools in the field of architectural design and construction. The keywords "Construction," "Construction Planning," and "Project Planning and Control" highlight the emphasis on efficient and effective planning of construction projects. Additionally, the presence of "Information Theory" suggests that information management and communication within the context of construction projects are also considered within this cluster.

Cluster 2 (Green): This cluster has the items with the highest occurrences. The keyword with the highest occurrence in this cluster and on the whole map is "Project Planning" with a total of 33 occurrences. It is closely followed by "Project Management" with 26 occurrences. Other keywords include "Decision Making", "Construction Industry", and "Automation" with 12. The keyword "Human" with 6 occurrences, is the least mentioned. Looking at those keywords, it suggests a strong emphasis on project planning, project management, decision making, and the construction industry within this cluster. The high occurrence of "Project Planning" implies that there is significant research interest in understanding and improving the planning processes of various projects. This is a Furthermore, the presence of "Project Management" highlights the focus on effectively managing projects and overseeing their execution. The occurrence of "Decision Making" indicates the exploration of decision-making processes within project contexts. Additionally, the keywords "Automation" and "Construction Industry" suggest a connection between automation technologies and their application within the construction industry. This implies a focus on the automation process to improve efficiency and productivity within construction projects. The occurrence of the keyword "Human" suggests that there may be research investigating the human aspect of project management or the interaction between human factors and project outcomes. To conclude this cluster emphasizes on the project planning and management aspects, with the integration of automation.

Cluster 3 (Blue): Overall this cluster has some midsized items In this cluster, the keyword with the highest occurrence is "Risk Management" with a total of 12 mentions. It is closely followed by "Information Management" and "Risk Assessment," each with 10 occurrences. Other significant keywords include "Big Data" with 6 occurrences and "Risk Perception" with 5 occurrences. These keywords suggest a focus on the management, assessment, and perception of risks within the analysed literature. The high occurrence of "Risk Management" indicates that there is significant research interest in understanding and mitigating risks associated with various contexts. Furthermore, the occurrence of "Information Management" suggests the importance of effective management and utilization of information in dealing with risks. It implies a focus on how information is collected, stored, and analysed to support risk management processes. Additionally, the presence of "Risk Assessment" highlights the evaluation and identification of potential risks in different domains or industries. This suggests a focus on assessing the likelihood and impact of risks to make informed decisions. The keyword "Big Data" suggests a potential interest in utilizing large volumes of data for risk analysis and decision-making processes. It implies a focus on leveraging data analytics and computational techniques to extract valuable insights from complex datasets. The occurrence of "Risk Perception" indicates the exploration of how individuals perceive and interpret risks, which can influence decision-making processes and risk management strategies. In conclusion, this cluster demonstrates a significant research interest in understanding and mitigating risks through effective risk management

and sufficient information. On the other hand the keyword of “Big Data” shows the connection of this cluster to the Artificial Intelligence aspect of the papers.

Cluster 4 (Yellow): This cluster has the second highest occurrence of a keyword, namely with the keyword Artificial Intelligence, with 27 occurrences. It indicates a strong focus on exploring the application and impact of Artificial Intelligence (AI) within the analysed literature. This is a explainable occurrence, since one of the main search terms of the selected papers was Artificial Intelligence. Furthermore, the occurrence of "Decision Support Systems" suggests a focus on the development and utilization of systems that assist decision-making processes, particularly in the context of construction projects or related domains. The keywords "Cost Estimating" and "Machine Learning" appear with 6 occurrences each. This implies that there may be research exploring the use of machine learning techniques for cost estimating purposes. Machine learning algorithms can be employed to analyse historical cost data and make predictions or improve accuracy in cost estimation processes. Altogether, this clusters indicates the presences of the third concept presented in Table 11.

Cluster 5 (Purple): In this cluster, the keyword with the highest occurrence is "Software Design" with a total of 8 mentions. It suggests a focus on the design and development of software systems within the analysed literature. Furthermore, the occurrence of "Application Programs" also relates to this mentioned theme. This focuses more on the software applications within the context of the researched papers. It implies a focus on understanding their functionalities, usage, or potential impact. The keyword "Human Resource Management" appears with 6 occurrences. This suggests potential research examining the management of human resources within the context of project planning. This keyword also indicates the not AEC industry related aspects of the papers, but the industries that also have common ground with Project Planning and Artificial Intelligence. This keyword implies a focus on understanding the role of human resources and their impact on project outcomes. Overall, the analysis of the keyword occurrences in this cluster emphasizes a focus on software design and the exploration of application programs. The presence of human resource management indicates potential research examining the management of human resources within project-related contexts.

Cluster 6 (Aqua): In this cluster, the keyword with the highest occurrence is "Scheduling" with a total of 11 mentions. This is explainable since project planning and scheduling are closely related to each other. The Project Management Institute (PMI) quotes: “The process of planning primarily deals with selecting the appropriate policies and procedures in order to achieve the objectives of the project. Scheduling converts the project action plans for scope, time cost and quality into an operating timetable.” (Moylan, 2002). Furthermore, the occurrence of "Budget Control" indicates potential research investigating methods and strategies for effectively managing and controlling project budgets. It implies a focus on understanding the financial aspects within project planning. The keyword "Genetic Algorithms", with 5 occurrences, suggests potential research exploring the application of genetic algorithms. These are computational optimization techniques inspired by natural selection (Muthulakshmi, 2019). Genetic algorithms can be utilized for solving complex scheduling or resource allocation problems (Arianyan et al., 2012; Pongcharoen et al., 2004). Overall, the analysis of the keyword occurrences in Cluster 6 highlights a focus on scheduling and the exploration of budget control and the application of genetic algorithms within project management contexts. This is part of the

The strong link between Artificial Intelligence and Decision Support Systems, with a link strength of 4.53, suggests a significant association between these two concepts within the analysed literature. This implies that research in the field of artificial intelligence often focuses on the development and application of decision support systems, indicating a mutual dependency and integration of these areas. The second strongest link between Project Planning and Project Management, with a link strength of 4.35, highlights the close relationship and interdependency between these two key aspects

of project management. This suggests that effective project planning plays a crucial role in successful project management outcomes, reinforcing the importance of strategic planning processes in achieving project objectives. The strongest link between keywords of different clusters, specifically Artificial Intelligence and Project Planning, with a link strength of 2.93, indicates a connection between the utilization of artificial intelligence techniques and their application in project planning processes. This link strengthens the observation made in the previous analysis that artificial intelligence has a significant influence on project-related aspects, extending its impact beyond decision support systems and integrating with project planning activities.

Considering the total link strength of the map being 135.50 and the 263 links in the entire map, these strong link strengths indicate a substantial amount of associations and connectivity within the analysed literature. This suggests an interconnected research landscape of the analysed papers, incorporating themes such as artificial intelligence, decision support systems, project planning, and project management. Altogether, these findings suggest that the integration of artificial intelligence and decision support systems can enhance project planning and project management practices. This indicates potential opportunities for leveraging advanced technologies and computational approaches improving decision-making processes, optimising resource allocation, and mitigating risks within a project context.

Artificial Intelligence

The bibliometric map, shown in Figure 27, is based on the articles focusing on Artificial Intelligence. Within this map five distinct clusters can be found. In general the words that have the most occurrences are the words, Artificial Intelligence, Project Planning, and Decision Support Systems. These keywords with its occurrences and all other keywords and occurrences can be found in Table 12. Apart from the five clusters some other specifics of the map can be named. The total number of keywords are 36, the total links are 270, and lastly the total link strength is 104. These numbers result in an average link strength per link of 0.39 and on average 7.5 links per keyword. Comparing this to the results of the Project Planning bibliometric map, it can be concluded that the Project Planning map has stronger links and more links per keyword. In the Artificial Intelligence map has a more centred keyword with strong links and not multiple like the other map has.

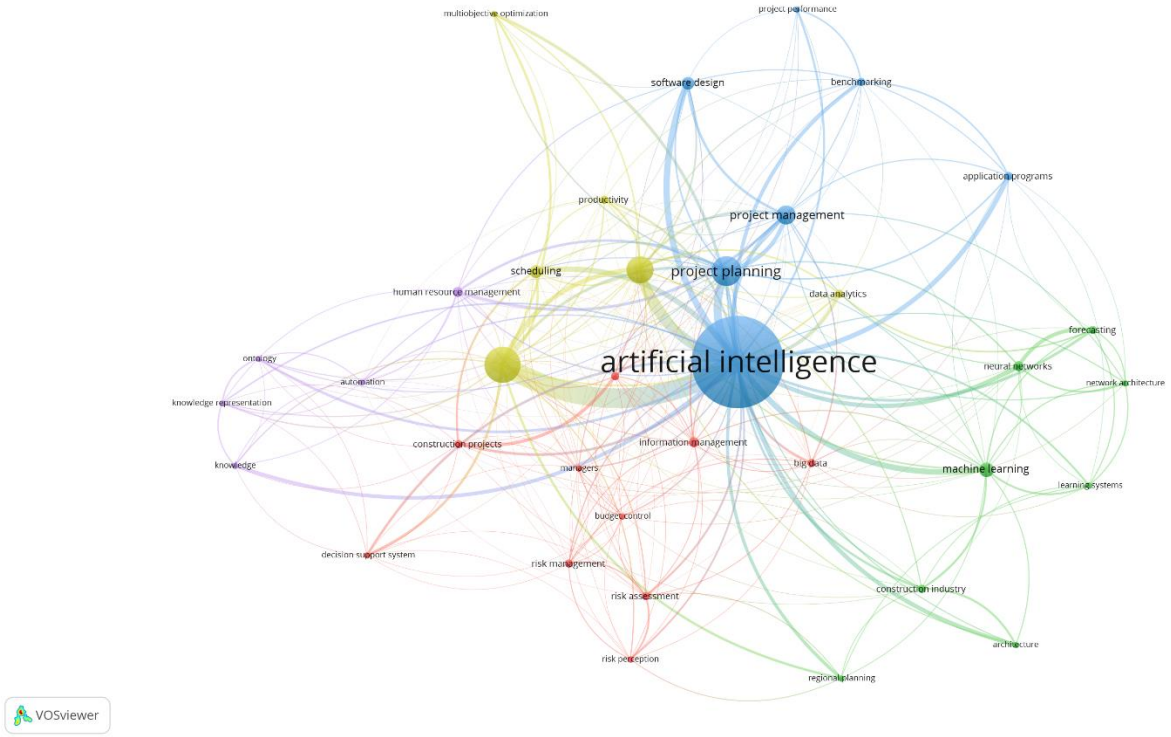


Figure 27 Bibliometric map of Keywords of Artificial Intelligence articles based on occurrences

The way this bibliometric map is analysed is the same as for the previous map. First, the different clusters present in the map, in this case five, will be analysed. These five clusters will be discussed one by one, to show the different themes of the clusters and the way they are structured. After looking at each individual cluster, the overall map is analysed. This will focus more on the link strength between keywords, but also the connection of the clusters between each other.

Table 12 Artificial Intelligence keywords map with occurrence per keyword

Cluster 1 *Red*		**Cluster 2** *Green*		**Cluster 3** *Blue*		**Cluster 4** *Yellow*		**Cluster 5** *Purple*	
Information Management	5	Machine Learning	7	Artificial Intelligence	43	Decision Support Systems	17	Human Resource Management	5
Big Data	4	Neural Networks	5	Project Planning	14	Decision Making	13	Automation	3
Construction Project	4	Construction Industry	4	Project Management	9	Scheduling	6	Knowledge	3
Risk Assessment	4	Forecasting	4	Software Design	6	Data Analytics	4	Knowledge Representation	3
Risk Management	4	Architecture	3	Application Programs	4	Productivity	4	Ontology	3
Uncertainty Analysis	4	Learning Systems	3	Benchmarking	4	Multiobjective Optimization	3		
Budget Control	3	Network Architecture	3	Project Performance	3				
Decision Support System	3	Regional Planning	3						
Managers	3								
Risk Perception	3								

Cluster 1 (Red): This cluster is the largest based on the number of items. However, based on occurrence this is not the case. This clusters has an emphasis on managing and utilizing information. The occurrence of "Information Management" suggests a focus on effectively managing project information to support the decision-making processes. The presence of "Big Data" highlights the importance of handling and analyzing large volumes of data. It suggests a potential interest in leveraging data analytics techniques to extract valuable insights and improve project outcomes. The occurrence of keywords such as "Construction Project," "Risk Assessment," "Risk Management," and "Uncertainty Analysis" indicates potential research exploring risk assessment methodologies, risk management strategies, and uncertainty analysis techniques. Additionally, the occurrence of "Budget Control" suggests a consideration of managing and controlling project budgets effectively. The presence of "Decision Support System" indicates potential research investigating the development and use of decision support systems to enhance project decision-making processes. The occurrence of "Managers" suggests a focus on the role of project managers and their influence on project outcomes. The presence of "Risk Perception" indicates the exploration of how individuals perceive and interpret risks within construction projects, highlighting the importance of understanding stakeholders' perspectives in managing project risks. Overall, the analysis of the first cluster suggests a focus on information management, big data, risk assessment, risk management, and decision support within the context of construction projects.

Cluster 2 (Green): In this cluster keywords focused on machine learning and the implementations of this are present. The occurrence of "Construction Industry" indicates a consideration of the unique characteristics and challenges associated with the construction sector. Therefore, in this cluster the combination of the machine learning aspects within the AEC industry is covered. The occurrence of "Machine Learning" already suggests research interest to use machine learning algorithms and techniques to improve various aspects of processes within the AEC industry. It implies a focus on using

algorithms and models to analyse construction data, improve decision-making processes, and optimize various aspects of construction projects. The presence of "Neural Networks" highlights a specific type of machine learning approach that mimics the structure and functioning of the human brain (Kanwisher et al., 2023). This suggests potential research exploring the application of neural networks for tasks such as pattern recognition, data analysis, decision-making, and the possible implementation in the AEC industry. However, this implies a focus on understanding the industry's specific requirements and exploring new approaches and solutions to address them. Besides “Neural Networks” the keywords "Forecasting," "Learning Systems," and "Network Architecture" further relate to machine learning, with "Forecasting" focusing on data-driven predictive modelling (Sangrody et al., 2018). "Learning Systems" indicates the exploration of adaptive or intelligent systems that improve decision-making processes, project performance, and resource allocation (Shakhsi-Niaei & Iranmanesh, 2015). "Network Architecture" suggests consideration for the design and optimization of different networks and the integration of digital systems and technologies within these networks. With regards to the keyword "Architecture," it can be interpreted in two possible meanings: architectural design and system architecture. However, in the context of the bibliometric map shown in Figure 27, its association with terms like "Artificial Intelligence," "Decision Making," and "Machine Learning" indicates that it is referring to system architecture rather than architectural design. This suggests a research interest in developing architecture-specific systems, models, or frameworks that integrate machine learning and advanced technologies.

Cluster 3 (Blue): This cluster demonstrates an emphasis on artificial intelligence (AI) within the analysed papers, which is also the main topic of this bibliometric map. Within the map, this cluster has the keyword with the highest occurrences of all the clusters, namely “Artificial Intelligence” . The occurrence of "Artificial Intelligence" suggests a significant research interest in understanding and exploring the application of AI techniques and methodologies. The presence of "Project Planning" indicates a focus on planning processes within the linked keywords. It suggests research topics about developing effective project plans, considering various factors, and optimizing project outcomes. Furthermore, the occurrence of "Project Management" highlights the importance of efficient project management practices, ensuring successful project execution, monitoring, and control. The inclusion of "Software Design" suggests research investigating the design and development of software systems specific to construction projects. The presence of "Application Programs" indicates potential research exploring specific programs or software applications utilized in the construction industry. It suggests a focus on understanding the functionalities, usage, or potential impact of such programs in project management. The occurrence of "Benchmarking" suggests potential research investigating benchmarking practices within the construction industry. It implies a focus on comparing project performance, processes, or outcomes with industry standards to identify areas for improvement. The presence of "Project Performance" highlights the evaluation and assessment of project outcomes and performance indicators. It suggests potential research examining factors that contribute to successful project performance and the identification of strategies for achieving desired project outcomes.

Cluster 4(Yellow): This cluster demonstrates a strong emphasis on decision support systems and decision-making processes. Furthermore, this cluster also has some high keyword occurrences. The occurrence of "Decision Support Systems" suggests research interest in developing and utilizing systems to assist decision-making processes, with a focus on leveraging technology and data-driven approaches to support informed decision making. The presence of "Decision Making" further emphasizes this. It suggests potential research exploring the factors influencing decision making, the utilization of decision models or frameworks, and strategies for effective decision making in a project. The occurrence of "Scheduling" indicates a consideration of scheduling processes within projects. Potential research about this topic will focus on scheduling methodologies, optimization techniques,

or technologies to enhance project planning and resource allocation. The inclusion of "Data Analytics" suggests a focus on utilizing data-driven approaches and techniques to extract valuable insights from construction project data. It implies research interest in data analysis, visualization, and interpretation to support project decision making, performance evaluation, or process improvement. The presence of keywords such as "Productivity" and "Multiobjective Optimization" indicates potential research exploring productivity enhancement strategies and optimization approaches within construction projects. It suggests a focus on improving efficiency, resource utilization, and achieving multiple objectives simultaneously within project environments.

Cluster 5(Purple): The last cluster has a focus on human resource management within the researched papers. The occurrence of "Human Resource Management" suggests research interest in understanding and improving the management of human resources within projects. It implies a focus on factors such as recruitment, training, performance evaluation, and retention of personnel in the construction industry. The presence of "Automation" highlights the consideration of automated systems or technologies. Potential research could explore the adoption and impact of automation on various aspects of projects, such as productivity, efficiency, and safety. The occurrence of "Knowledge" suggests a focus on the utilization and management of knowledge within construction projects. It implies potential research investigating knowledge acquisition, sharing, and utilization strategies within the construction industry. The inclusion of keywords such as "Knowledge Representation" and "Ontology" indicates a focus on representing and organizing knowledge in a structured manner within the construction domain. It implies potential research exploring knowledge representation techniques, ontologies, or semantic models to enhance knowledge management and sharing within construction projects.

The strongest link in this bibliometric map is between "Artificial Intelligence" and "Decision Support Systems," with a link strength of 6.24. This suggests a significant association between these two keywords within the analysed literature. The analysed papers highlight the significance of decision support systems in connection to AI, showing the interconnection between those topics. This suggests an interdependence between these two keywords. The second-highest link, between "Artificial Intelligence" and "Decision Making," with a link strength of 3.33, this link strength also strengthens the above mentioned clarification since “Decision Making” can be seen as a part of “Decision Support Systems”. Furthermore both “Decision Making” and “Decision Support Systems” are part of another cluster than "Artificial Intelligence". This emphasizes the strong connection between these two clusters (clusters 3 and 4). Next to this, the strongest link between keywords of the same clusters, specifically "Artificial Intelligence" and "Project Planning," with a link strength of 2.93, indicates a connection between the use of artificial intelligence techniques and their application in project planning processes. This link strengthens the observation made in the previous bibliographic analysis that artificial intelligence has a significant influence on project-related aspects.

These findings indicate a substantial amount of associations and connectivity within the analysed literature, considering the total link strength of 104 and the 270 links in the entire bibliometric map. The average link strength per link of 0.39 further indicates this as well, even though it is lower than in the first bibliographic map (Figure 27). This suggests an integrated research landscape, incorporating themes such as artificial intelligence, decision support systems, decision making, and their interplay with various clusters.

Appendix B: Explanation of ID clusters data

start of ID	explanation
A	Standard from Primavera
AB	All levels of effort
B(2digits)	West Design, Work Preparation, and Execution
BBR	Bearing Block Replacement
BF.CON	Contractor Buffer
B	Demolition or Temporary Placement
C	Basement Application or Demolition
CM	Milestones
DC	Deep Cut North
DCS	Deep Cut South
DPSM	Delivery Packages
E	(Installed) Utilities
EPCM.CM	Milestones (Mostly Finished)
EW	Early Works – Sheet Piling, Combiwall & Anchor Works
FP	Freeze Period Database Change
GK	Retaining Wall
H	Concrete Works – Deep Location
HST	Concrete Works – Moot 20
KW03	KW003 Execution (One of the Bridges)
KW002	KW002 Execution (One of the Bridges)
L3	Linking Detail Planning and Project Planning
M	General Activities
M-	Steps – Section Roadworks
M-TGK	Design of Retaining Walls
MTrBr	KW085 Execution
MPA	Start Build (Start Milestone)
NBB	Install New Bearing Blocks
OWN	Underlying Road Network (for Structure 2)
PCS	Pile Cap Strengthening
PMS-EXE-M	Start and Finish Milestones
Proc-M	Procurement – Central Area
Prod	Production Time for Prefabrication
PS	Strengthen Existing Pillars/Basement/Abutment
PT	Execution of Portals for DVM
RBP	RBP Buffer
RD_(HRL or HRR)	Roadworks West
ST	Standard from Primavera
STXML	General Activities, Mainly West
TGK	Temporary Sheet Piles (Temporary Ground Retaining)
TJS	Temporary Jacking Support
TPC-KKW	Sewer Pumping Station KK Road
TSS	Temporary Seepage Screen Design

VERG	Permit
VV	Auxiliary Bridge Steelworks or Reinforcement Concrete Works for Structure 1
W	(Install) Cable, Utilities
W-C	Steps – Roadworks Section C1
WC	Water Compensation

Appendix C: Final generalized code Clustering

```
# %%
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD
from sklearn.cluster import Kmeans
import matplotlib.pyplot as plt
import joblib
from sklearn.metrics import silhouette_score


# %%
# Data that needs to be clustered, make sure that the column with the
activities names in there is called 'Name'
data = pd.read_excel(r" ") #insert filepath to data file

# %%
data['lemmed'] = data['lemmed'].str.replace(',', '') #make sure that the
column with the data is called lemmed (this is the case if the data cleaning
is correctly followed)

texts = data["lemmed"].fillna("")

# TfidfVectorizer is used because it can be used on complex data, it captures
the importance of terms while considering their frequency in the context of
the entire corpus. This helps in distinguishing between common words and terms
that carry specific meaning within individual documents
vectorizer = TfidfVectorizer()

vectors = vectorizer.fit_transform(texts)

# Make the vectorization 98wardin
n_components = 2  # Number of dimensions for visualization, 2 is chosen to
make an X- and Y-axis
svd = TruncatedSVD(n_components=n_components)
vectors_reduced = svd.fit_transform(vectors)

# %%
# List to store silhouette scores for different cluster numbers
silhouette_scores = []

# Set a range of cluster numbers to try. Set between 200 - 400 based on
previously manually made clusters. The range can be adapted based on the size
of the planning put in.
Min_clusters = 200
max_clusters = 400

# Iterate over different cluster numbers
```

```
for num_clusters in range(min_clusters, max_clusters + 1):
    kmeans = Kmeans(n_clusters=num_clusters, random_state=0)
    labels = kmeans.fit_predict(vectors_reduced)
    silhouette_avg = silhouette_score(vectors_reduced, labels)
    silhouette_scores.append(silhouette_avg)

# Plot silhouette scores. This way the 99wardin score can be chosen for the
performance of the K-means clustering
plt.figure(figsize=(10, 6))
plt.plot(range(min_clusters, max_clusters + 1), silhouette_scores, marker='o')
plt.xlabel('Number of Clusters')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Score for Different Cluster Numbers')
plt.show()


# %%
# Put in the number with the 99wardin 99warding99e score (or another
99warding99 number of clusters)
n_clusters = []#add amount from 99warding99e score

# %%
# Initialize and train the K-Means model
kmeans = Kmeans(n_clusters=n_clusters, random_state=0)
cluster_labels = kmeans.fit_predict(vectors_reduced)

data_cluster_mapping = {text: label for text, label in zip(texts,
cluster_labels)}

# Create a dictionary to map data inputs to cluster labels
joblib.dump(kmeans, 'kmeans_model_final.pkl') #needed for the labelling
joblib.dump(vectorizer, 'tfidf_vectorizer_final.pkl') #needed for the
labelling
joblib.dump(svd, 'truncated_svd_final.pkl') #needed for the labelling

cluster_labels_df = pd.DataFrame({'Cluster Labels': cluster_labels})

# Save the cluster labels to a CSV file
cluster_labels_df.to_csv('cluster_labels_final.csv', index=False)

# Export the cluster labels to a new column in the original DataFrame
data['Cluster'] = cluster_labels

# Save the original DataFrame with cluster labels to a new Excel file
data.to_excel('clustered data .xlsx', index=False) #the file is saved with the
name Clustered data
```

Listing 7 Final generalized code Clustering

Appendix D: Data preparation code

Part of Speech

```
from nltk.tokenize import word_tokenize
from nltk.tag import pos_tag

df['Tokenized_Name'] = df['Name'].apply(lambda x: word_tokenize(x))
df['POS_Tags'] = df['Tokenized_Name'].apply(lambda x: pos_tag(x))

# Function to extract verbs and nouns for each row
def extract_verbs_nouns(tags):
    verb_columns = [f'Word_{i}' for i, pos_tag in enumerate(tags) if
pos_tag[1] in ['VB', 'VBD', 'VBG', 'VBN', 'VBP', 'VBZ']]
    noun_columns = [f'Word_{i}' for i, pos_tag in enumerate(tags) if
pos_tag[1] == 'NN']
    return verb_columns, noun_columns

df[['Verb_Columns', 'Noun_Columns']] = df['POS_Tags'].apply(lambda x:
pd.Series(extract_verbs_nouns(x)))

# Flatten the lists in 'Verb_Columns' and 'Noun_Columns'
df['Verb_Columns'] = df['Verb_Columns'].apply(lambda x: ', '.join(x) if x else
'')
df['Noun_Columns'] = df['Noun_Columns'].apply(lambda x: ', '.join(x) if x else
'')

# Create a new DataFrame with 'ID', 'Name', 'Verb_Columns', and 'Noun_Columns'
df_selected = df[['ID', 'Name', 'Verb_Columns', 'Noun_Columns']]
```

Listing 8 identify PoS and extracting nouns and verbs

Text between brackets and punctuation marks

```
def remove_text_in_brackets(text):
    return re.sub(r'\([^)]*\)', '', text)

df['Name'] = df['Name'].astype(str).apply(remove_text_in_brackets)

def remove_punctuation(text):
    return re.sub(r'[^a-zA-Z0-9\s]', '', text)

df['Name'] = df['Name'].astype(str).apply(remove_punctuation)
```

Listing 9 removal of punctuation marks and text between brackets

Stemming and Lemmatization

```
stemmer = PorterStemmer()

df['Stemmed'] = df['Name'].apply(lambda x: ' '.join([stemmer.stem(word) for
word in nltk.word_tokenize(x)]))
```

Listing 10 Stemming of words

```
nltk.download('wordnet')

lemmatizer = WordNetLemmatizer()

df['lemmed'] = df['Name].apply(lambda x: ' ' .join([lemmatizer.lemmatize(word)
for word in nltk.word_tokenize(x)]))
```

Listing 11 Lemmatization of words

Translation of Dutch words

```
nlp = spacy.load("en_core_web_lg")
nlp.add_pipe("language_detector")

def clean(sentence):
    cleaned_text = list()
  text = sentence.split() #sentence corresponds to the content/sentence of a
cell in the column 'Name'; text then contains a list of terms within the
sentence
    for term in text:
        doc = nlp(term)
        if(doc._.language == 'en'):
            cleaned_text.append(doc.text)
    return " ".join(cleaned_text)
```

Listing 12 Removal non-English words

Project specific terms

```
#removal of words with digits
def remove_words_with_digits(text):
    return re.sub(r'\b\w*\d\w*\b', '', text)

df['Name'] = df['Name'].astype(str).apply(remove_words_with_digits)

#removal of words that indicate a direction
directions = ["north", "south", "east", "west", "northeast", "northwest",
"southeast", "southwest", "middle"]

pattern1 = r'\b(?:' + '|'.join(directions) + r')\b'

def remove_directions(text):
    return re.sub(pattern1, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_directions)

#removal of words that indicate a compartment of the project
components = [ "part", "section", "component", "element", "segment", "level",
"axis", "phase", "sub"]

pattern2 = r'\b(?:' + '|'.join(components) + r')\b'

#removal of words that indicate a count
```

```
def remove_components(text):
    return re.sub(pattern2, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_components)

pattern3 =
r'\b(?:first|second|third|fourth|fifth|sixth|seventh|eighth|ninth|tenth)\b'

def remove_pattern3(text):
    return re.sub(pattern3, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_directions)
```

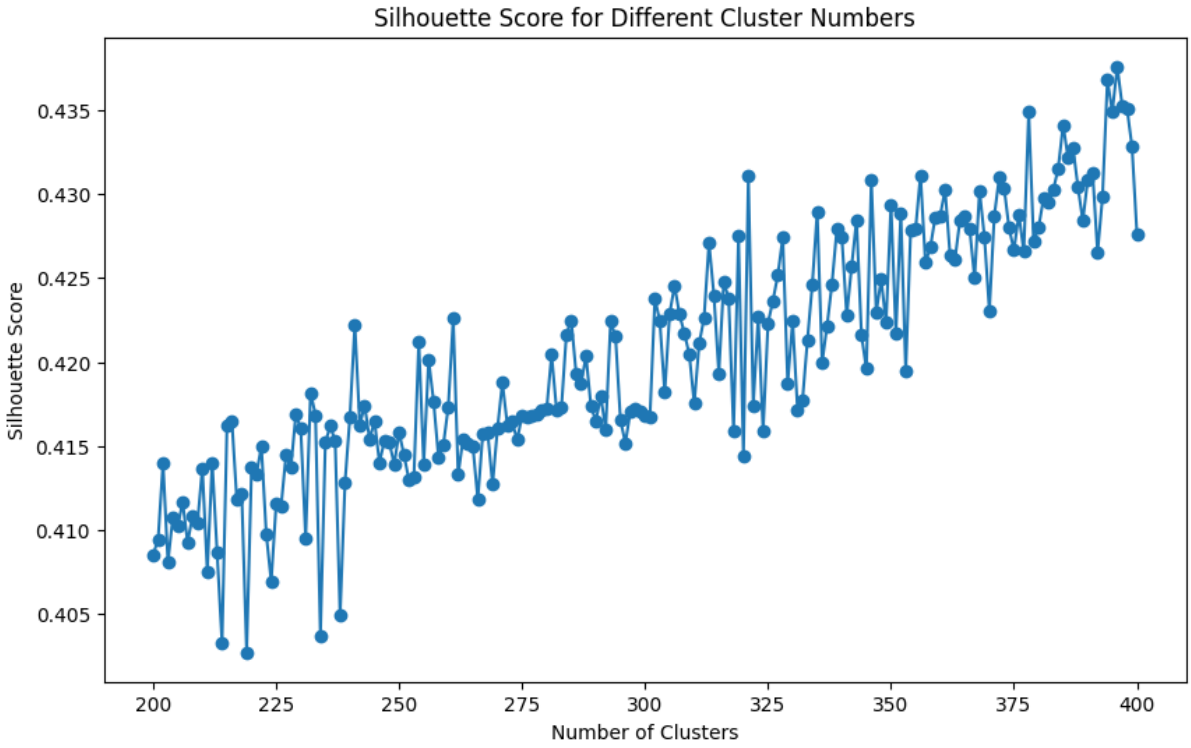
Listing 13 Categories of removal of project specific terms

```
#removal of residual abbreviations
abbriviations = "NV", "LS", "SG", "A", "B", "MS", "TC", "HS", "WT", "HD", "C",
"D", "E", "RP", "LD", "NV", "NW", "NE", "KWW", "W", "SN", "NAP", "UWC", "HWA",
"GEWI", "m", "ID", "DP", "NS", "km", "tbv", "HRL", "HRR", "NSA", "KZNPD",
"AMSWG", "DDI", "ALAP", "HWN", "GWW", "BLVC", "H", "incl", "KZN", "UPS",
"KKW", "L", "j", "VERG", "SHO", "ZOAB", "F", "WLS", "SCVCAUO", "PDR", "TBGN",
"ab", "VV", "RFP", "ITP", "GK", "VERG", "RWS", "sh", "AMS"]

pattern3 = r'\b(?:' + '|'.join(abbriviations) + r')\b'

def remove_abbriviations(text):
    return re.sub(pattern3, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_abbriviations)

#removal of residual words
other_words = "end", "days", "expired", "ramp", "closure", "Type",
"deprecated", "noniFAT"]

pattern4 = r'\b(?:' + '|'.join(other_words) + r')\b'

def remove_other_words(text):
    return re.sub(pattern4, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_other_words)
```

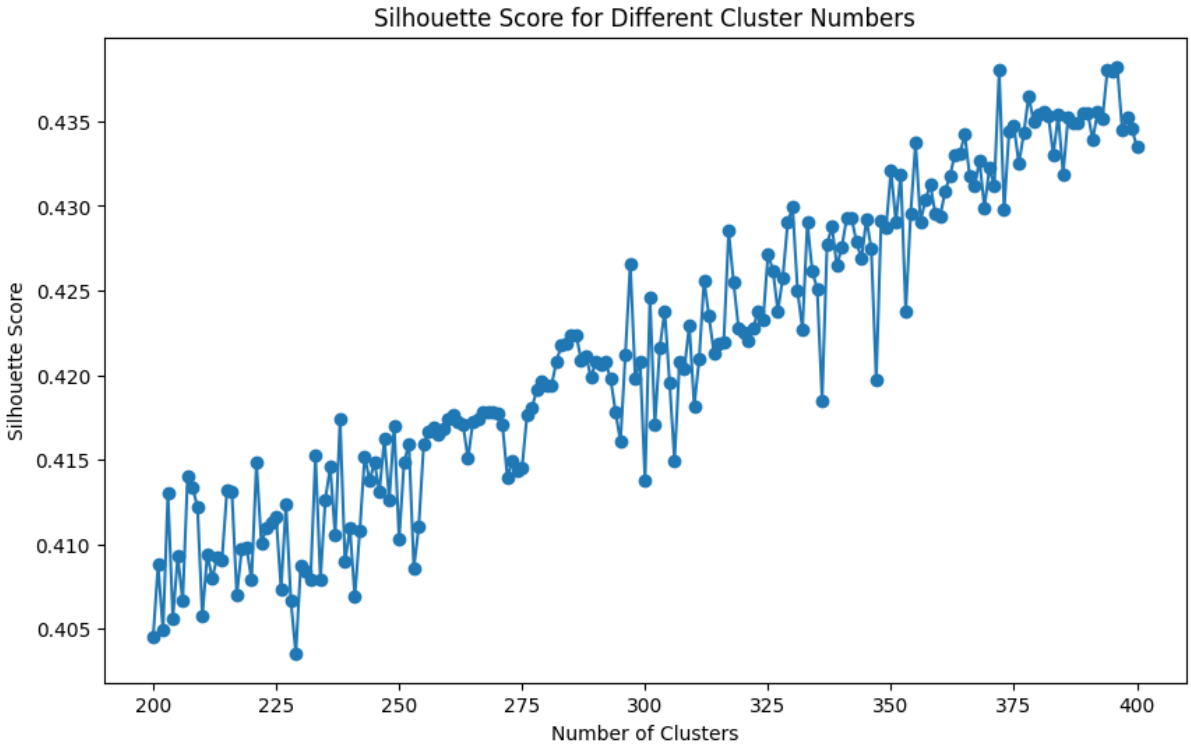
Listing 14 Removal residual abbreviations and word

Appendix E: Silhouette scores of different data preparation methods

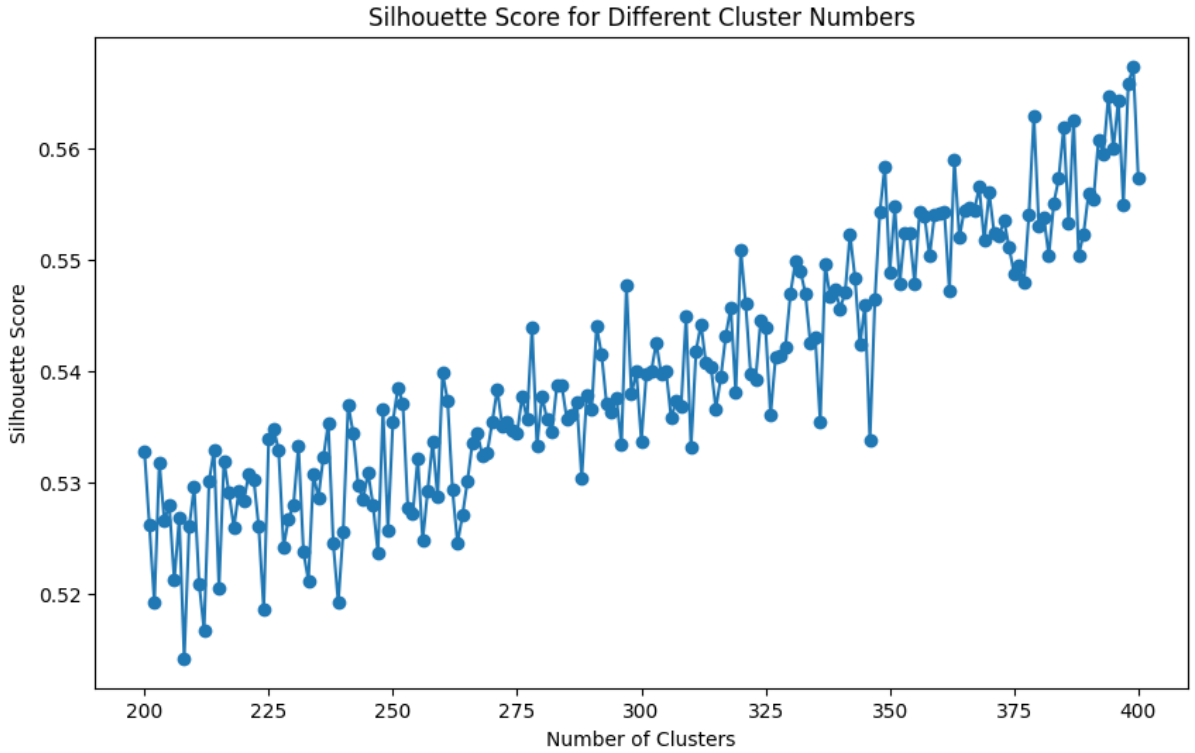
Original data, K = 396



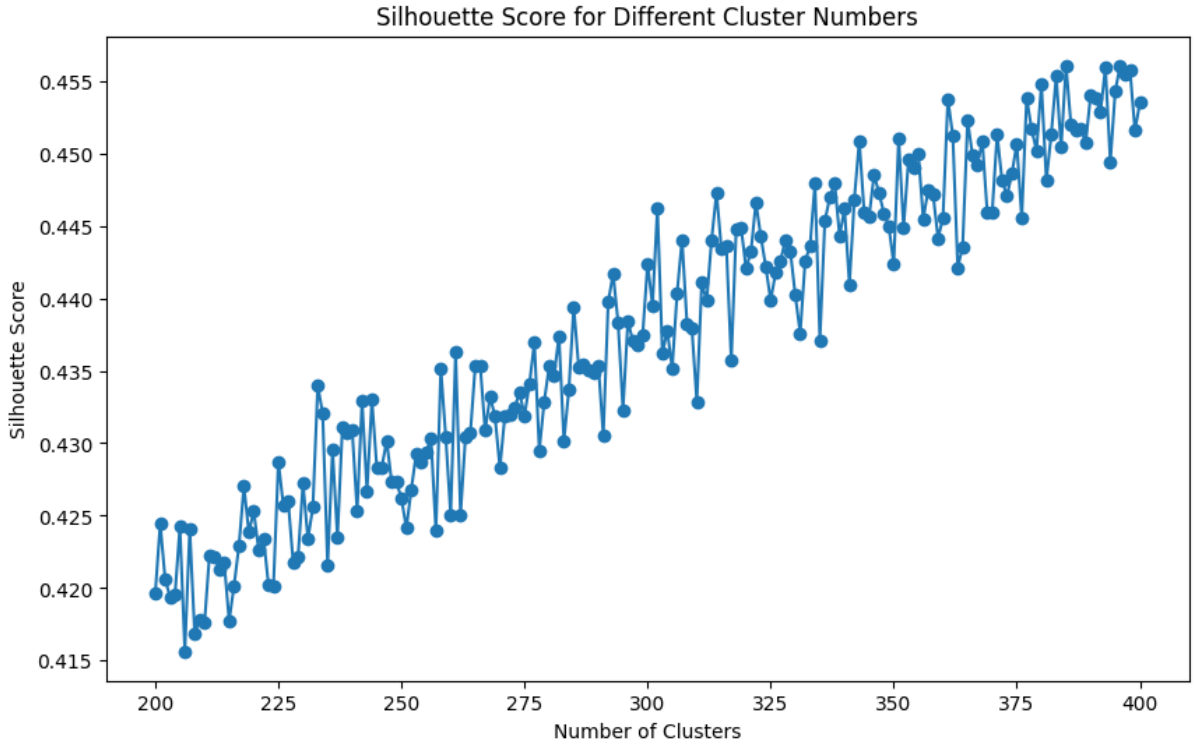
Stemmer, K = 396



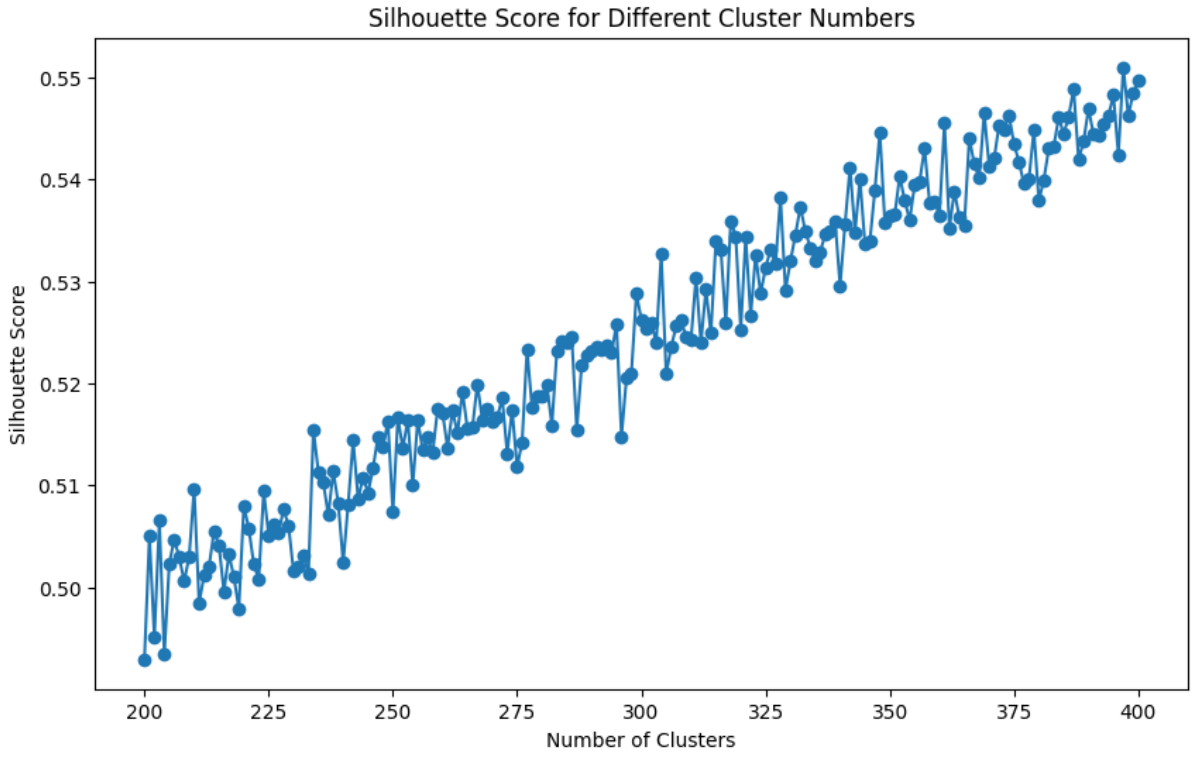
Part of Speech, K = 399



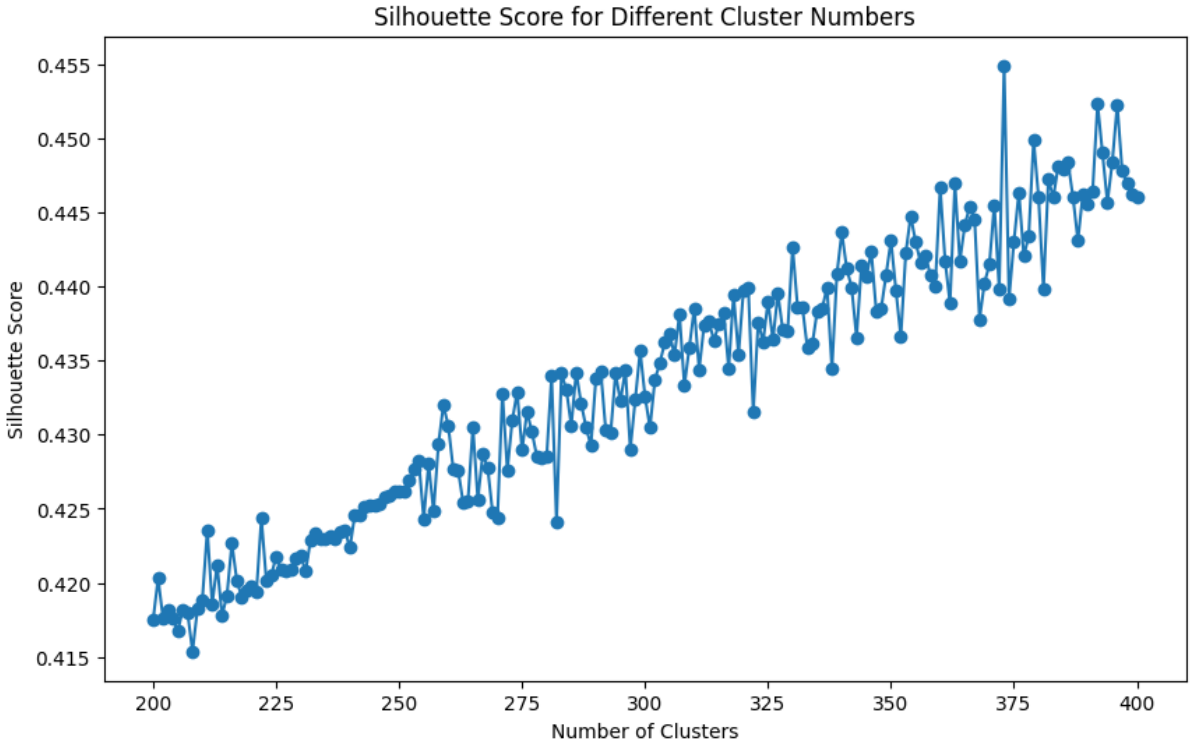
Removal of punctuation, K = 385



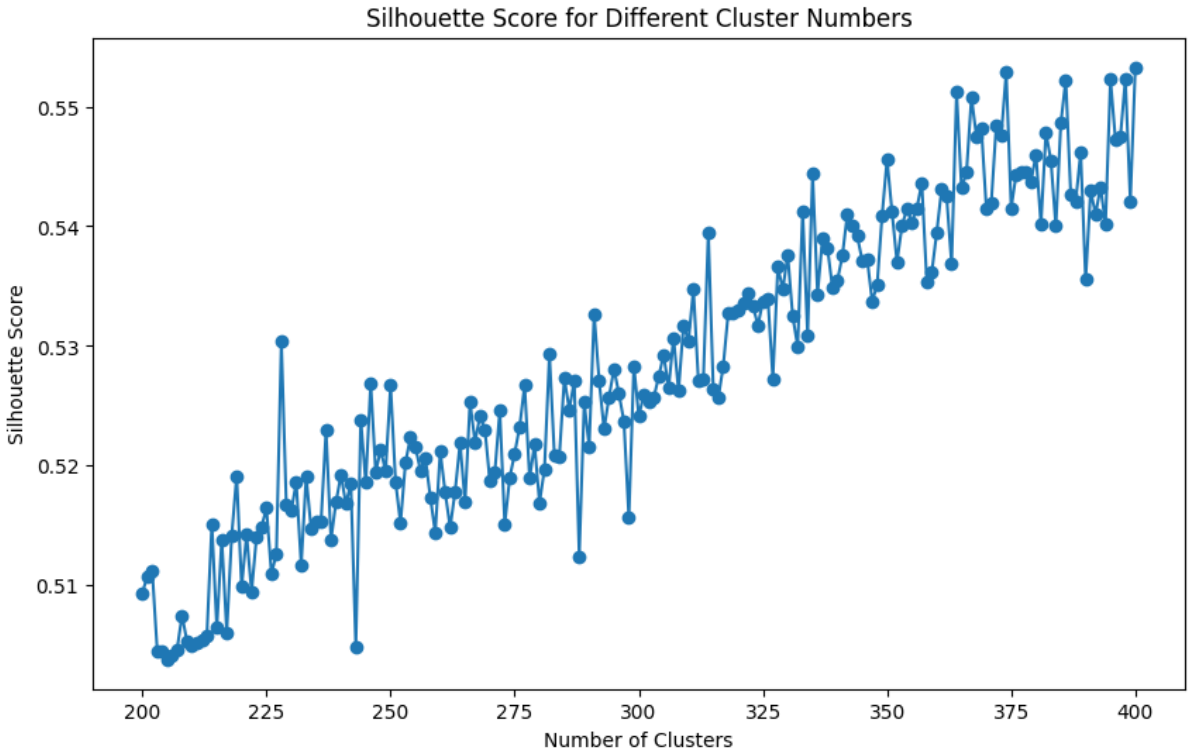
Project specific terms, K = 397



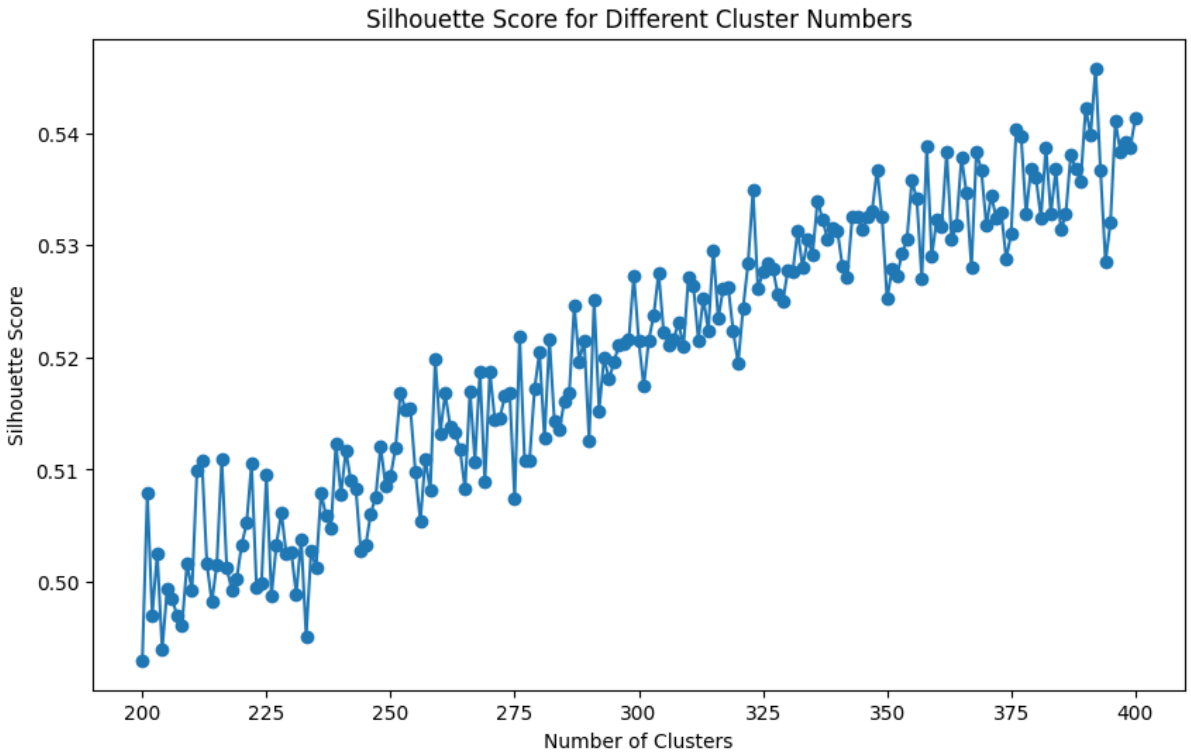
Removal of Dutch words, K =



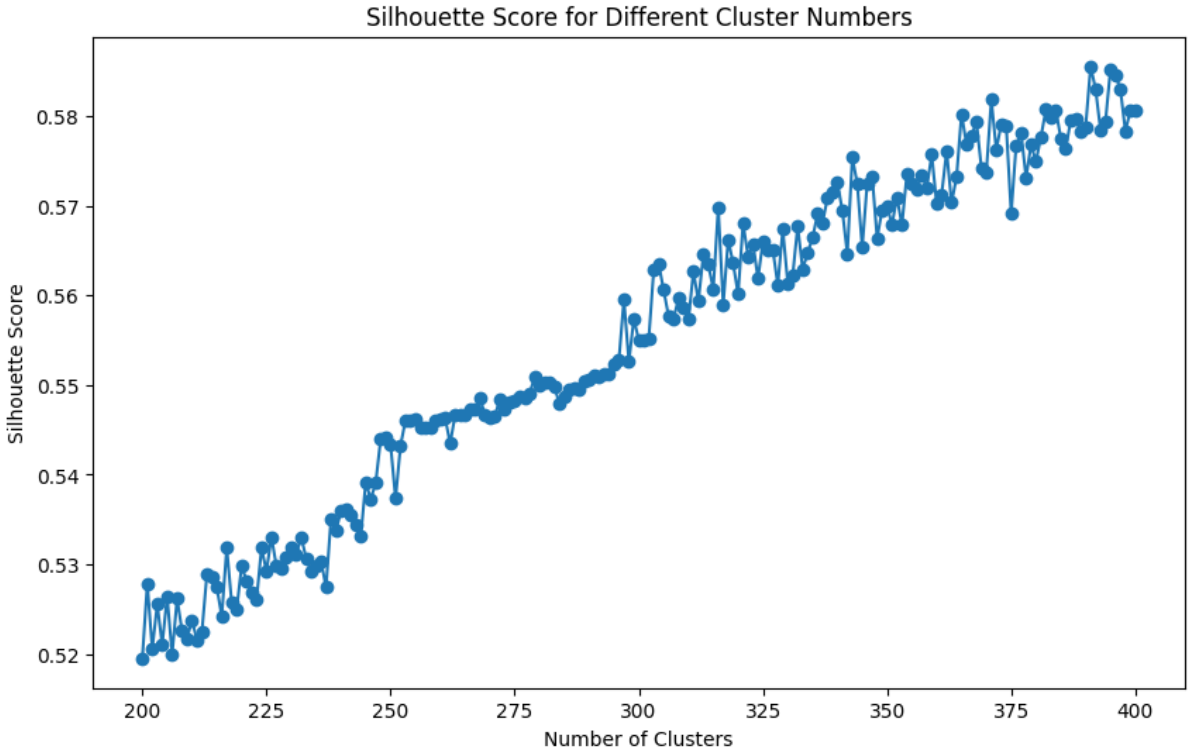
All cleaning (behalve dutch), K = 400



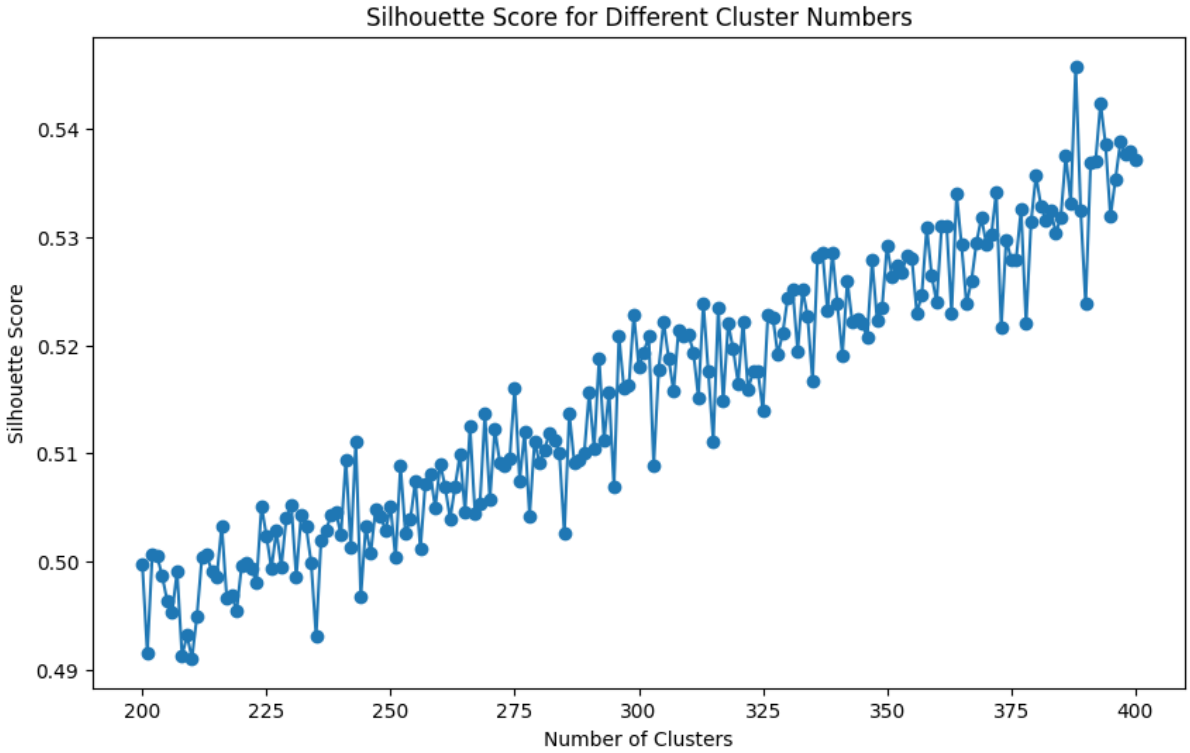
All positive adaptions, K = 392



All, K = 392



All (other cleaning sequence), K = 388



Appendix F: Final generalized code Labelling

```
 %%
import pandas as pd
import joblib
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD

# %%
# Load cluster labels from the CSV file
cluster_labels = pd.read_csv(r" ") #instert previously clustered file


# %%
# Load new construction planning dataset
new_data = pd.read_excel(r" ") #insert filepaht of the new planning that will
be labelled

# %%
# Load the Truncated SVD model, this is the file with the clusters in there
that are made with the use of K-means

svd = joblib.load(r" ") #filepath from truncated_svd_final.pkl which is
defined in the clustering code

num_components = svd.n_components

print(f"Number of components in the loaded model: {num_components}")

# %%
# Load the pre-trained K-Means model from training dataset
kmeans = joblib.load(r" ") #filepath from kmeans_model_final.pkl which is
defined in the clustering code

# Use the same vectorizer as used for clustering, to ensure coherent outcome
vectorizer = joblib.load(r" ") #filepath from tfidf_vectorizer_final.pkl which
is defined in the clustering code

# %%
# Fill missing values in the "Name" column with an empty string
new_data['lemmed'].fillna('', inplace=True) #make sure that the column with
the data is called lemmed (this is the case if the data cleaning is correctly
followed)

# using the loaded vectorizer to transform the data
name_vectors = vectorizer.transform(new_data['lemmed'])

# %%
# Apply Truncated SVD to match the dimensionality
```

```
name_vectors_reduced = svd.transform(name_vectors)

# Predict cluster labels for the new data
cluster_labels_new = kmeans.predict(name_vectors_reduced)

# Assign cluster labels to the new_data DataFrame
new_data['Cluster'] = cluster_labels_new

# %%
new_data.to_excel('new_data_with_labelled', index=False) #the file is saved
with the name new_data_with_labelled
```

Listing 15 Final generalized code Labelling

Appendix G: Outcome different preparations methods

The results of the two analysation methods can be seen in this Appendix. In this appendix, the attributes, as well as advantages and disadvantages of both evaluation methods are described. The use of cosine similarity and the silhouette score to evaluate clustering results in the context of data cleaning methods is chosen strategically. Since it shows how well the clusters are formed by two different evaluation methods. When understanding the pros and cons of these both methods the scores can also be explained better. This two-way evaluation method is useful in providing insight in the complexity involved with the selection of data preparation procedures.

Cosine similarity score

Cosine similarity, a vector space measure, is used in this research to determine the semantic similarities of activity names within the different clusters. It calculates the similarity between two vectors in a (multi-)dimensional space (Kitasuka et al., 2012). This metric is employed to assess the semantic similarity between items, which can include text documents, words, or other data points. The fundamental concept behind cosine similarity is the calculation of the cosine of the angle between these vectors, which serves as a measure of their similarity (Rizqi Lahitani et al., n.d.). The process involves representing each item as a vector in a multi-dimensional space, where each dimension corresponds to different features or attributes of the items. The resulting cosine similarity value ranges from -1 to 1, where a value of 1 signifies perfect similarity, 0 indicates no similarity, and -1 reflects perfect dissimilarity. Furthermore, cosine similarity can be applied to evaluate the impact of data cleaning methods on preserving semantic relationships within a dataset (Chaudhuri et al., 2006). A high cosine similarity score after data cleaning indicates that the data cleaning process has succeeded in maintaining the semantic coherence of items within the same cluster.

Therefore, the use of cosine similarity aligns with the aim of measuring the degree of semantic similarities between activity names within a cluster. Since the clustering of activity names is based on their underlying terminology, this measure is a good option. However, despite the presence of additional data cleaning methods or modifications to sequencing, the increase in cosine similarity scores remains rather limited. The complexity of activity names and the size of the dataset might clarify this occurrence.

For conducting the analysis among the different datasets, a code was developed to calculate the similarity score for each cluster subsequent to executing K-means clustering. Following this computation, a box plot is generated in Microsoft Excel.

```
activities = df['Name'].fillna('').tolist()

# Calculate average pairwise cosine similarity within each cluster
for cluster in unique_clusters:
    cluster_indices = np.where(df['Cluster'] == cluster)[0]
    cluster_tfidf_matrix = tfidf_matrix[cluster_indices]

    # Calculate cosine similarity matrix
    similarity_matrix = cosine_similarity(cluster_tfidf_matrix)

    # Calculate average similarity score for the cluster
    average_score = np.mean(similarity_matrix)

    # Get the count of activities in the cluster
    activity_count = len(cluster_indices)

    # Store the results in the list
    cluster_results.append({'Cluster': cluster, 'AverageSimilarityScore':
average_score, 'ActivityCount': activity_count})

# Create a DataFrame from the list of results
result_df = pd.DataFrame(cluster_results)
```

Silhouette score

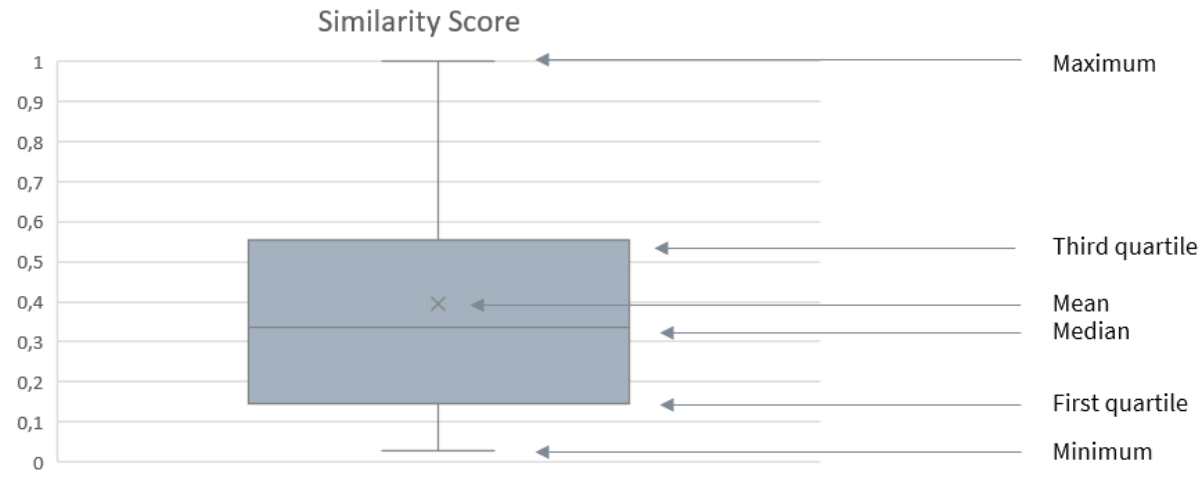
The silhouette score is a widely used cluster evaluation metric. It evaluates the degree of separation between clusters, thereby offering insights into the effectiveness of the clustering methodology in uniting similar activities while segregating dissimilar ones (Ogbuabor & F. N, 2018). The silhouette score is especially useful when dealing with an extensive amount of data. At its core, the Silhouette Score is determined by calculating a "silhouette" value for each data point in the dataset, assessing the data point's similarity to its own cluster in comparison to other clusters. These individual silhouette scores, which span from -1 to 1, are averaged to generate an overall Silhouette Score of each cluster.

Overview of pros and cons

In order to know what aspects of every tool are important, an overview is provided of the differences between the two scoring methods.

Table 13 Overview cosine similarity and silhouette score

Aspect	Cosine Similarity Score	Silhouette Score
Metric Type	Similarity metric based on vectors	Quality metric based on cluster structure
Application Focus	Semantic similarity within clusters	Overall cluster quality and separation
Strengths	Captures semantic similarity effectively	Measures cluster quality and separation effectively
	Suitable for assessing semantic cohesion	Identifies structural issues in clusters
	Useful for preserving semantic meaning	Offers insights into practical cluster separation
	Valuable for evaluating data cleaning impact	Detects issues related to cluster cohesion and separation
Weaknesses	Less informative about cluster separation	Less sensitive to semantic similarities
	Doesn't account for cluster size/distribution	Vulnerable to imbalanced cluster sizes
	Vulnerable to variations in naming conventions	Not specialized for semantic coherence
	May not fully represent overall cluster quality	Sensitive to the number/distribution of clusters
Decision Factors	Specific goals related to semantic coherence	Emphasis on clear cluster separation and quality
	Nature of the dataset and diversity of naming	Cluster distribution, size, and overall structural quality
	Expertise required in domain-specific context	Sensitivity to the number of clusters and their distributions



Original data

First to generate a baseline to compare the different data preparation techniques to, the original data is analysed. As the figure shows, the K will be determined by the silhouette score. For this data the K is set at 396. The outcome of this silhouette score computation, as well as the silhouette scores for the other tests in this chapter can be found in # %%

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD
from sklearn.cluster import Kmeans
import matplotlib.pyplot as plt
import joblib
from sklearn.metrics import silhouette_score


# %%
# Data that needs to be clustered, make sure that the column with the
activities names in there is called 'Name'
data = pd.read_excel(r" ") #insert filepath to data file

# %%
data['lemmed'] = data['lemmed'].str.replace(',', '') #make sure that the
column with the data is called lemmed (this is the case if the data cleaning
is correctly followed)

texts = data["lemmed"].fillna("")

# TfidfVectorizer is used because it can be used on complex data, it captures
the importance of terms while considering their frequency in the context of
the entire corpus. This helps in distinguishing between common words and terms
that carry specific meaning within individual documents
vectorizer = TfidfVectorizer()

vectors = vectorizer.fit_transform(texts)

# Make the vectorization 113wardin
n_components = 2  # Number of dimensions for visualization, 2 is chosen to
make an X- and Y-axis
svd = TruncatedSVD(n_components=n_components)
vectors_reduced = svd.fit_transform(vectors)

# %%
# List to store silhouette scores for different cluster numbers
silhouette_scores = []

# Set a range of cluster numbers to try. Set between 200 - 400 based on
previously manually made clusters. The range can be adapted based on the size
of the planning put in.
Min_clusters = 200
```

```
max_clusters = 400

# Iterate over different cluster numbers
for num_clusters in range(min_clusters, max_clusters + 1):
    kmeans = Kmeans(n_clusters=num_clusters, random_state=0)
    labels = kmeans.fit_predict(vectors_reduced)
    silhouette_avg = silhouette_score(vectors_reduced, labels)
    silhouette_scores.append(silhouette_avg)

# Plot silhouette scores. This way the 114wardin score can be chosen for the
performance of the K-means clustering
plt.figure(figsize=(10, 6))
plt.plot(range(min_clusters, max_clusters + 1), silhouette_scores, marker='o')
plt.xlabel('Number of Clusters')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Score for Different Cluster Numbers')
plt.show()


# %%
# Put in the number with the 114wardin 114warding114e score (or another
114warding114 number of clusters)
n_clusters = []#add amount from 114warding114e score

# %%
# Initialize and train the K-Means model
kmeans = Kmeans(n_clusters=n_clusters, random_state=0)
cluster_labels = kmeans.fit_predict(vectors_reduced)

data_cluster_mapping = {text: label for text, label in zip(texts,
cluster_labels)}

# Create a dictionary to map data inputs to cluster labels
joblib.dump(kmeans, 'kmeans_model_final.pkl') #needed for the labelling
joblib.dump(vectorizer, 'tfidf_vectorizer_final.pkl') #needed for the
labelling
joblib.dump(svd, 'truncated_svd_final.pkl') #needed for the labelling

cluster_labels_df = pd.DataFrame({'Cluster Labels': cluster_labels})

# Save the cluster labels to a CSV file
cluster_labels_df.to_csv('cluster_labels_final.csv', index=False)

# Export the cluster labels to a new column in the original DataFrame
data['Cluster'] = cluster_labels

# Save the original DataFrame with cluster labels to a new Excel file
data.to_excel('clustered data .xlsx', index=False) #the file is saved with the
name Clustered data
```

Listing 7 Final generalized code Clustering

Appendix D: Data preparation code

Part of Speech

```
from nltk.tokenize import word_tokenize
from nltk.tag import pos_tag

df['Tokenized_Name'] = df['Name'].apply(lambda x: word_tokenize(x))
df['POS_Tags'] = df['Tokenized_Name'].apply(lambda x: pos_tag(x))

# Function to extract verbs and nouns for each row
def extract_verbs_nouns(tags):
    verb_columns = [f'Word_{i}' for i, pos_tag in enumerate(tags) if
pos_tag[1] in ['VB', 'VBD', 'VBG', 'VBN', 'VBP', 'VBZ']]
    noun_columns = [f'Word_{i}' for i, pos_tag in enumerate(tags) if
pos_tag[1] == 'NN']
    return verb_columns, noun_columns

df[['Verb_Columns', 'Noun_Columns']] = df['POS_Tags'].apply(lambda x:
pd.Series(extract_verbs_nouns(x)))

# Flatten the lists in 'Verb_Columns' and 'Noun_Columns'
df['Verb_Columns'] = df['Verb_Columns'].apply(lambda x: ', '.join(x) if x else
'')
df['Noun_Columns'] = df['Noun_Columns'].apply(lambda x: ', '.join(x) if x else
'')

# Create a new DataFrame with 'ID', 'Name', 'Verb_Columns', and 'Noun_Columns'
df_selected = df[['ID', 'Name', 'Verb_Columns', 'Noun_Columns']]
```

Listing 8 identify PoS and extracting nouns and verbs

Text between brackets and punctuation marks

```
def remove_text_in_brackets(text):
    return re.sub(r'\([^)]*\)', '', text)

df['Name'] = df['Name'].astype(str).apply(remove_text_in_brackets)

def remove_punctuation(text):
    return re.sub(r'[^a-zA-Z0-9\s]', '', text)

df['Name'] = df['Name'].astype(str).apply(remove_punctuation)
```

Listing 9 removal of punctuation marks and text between brackets

Stemming and Lemmatization

```
stemmer = PorterStemmer()

df['Stemmed'] = df['Name'].apply(lambda x: ' '.join([stemmer.stem(word) for
word in nltk.word_tokenize(x)]))
```

Listing 10 Stemming of words

```
nltk.download('wordnet')

lemmatizer = WordNetLemmatizer()

df['lemmed'] = df['Name].apply(lambda x: ' ' .join([lemmatizer.lemmatize(word)
for word in nltk.word_tokenize(x)]))
```

Listing 11 Lemmatization of words

Translation of Dutch words

```
nlp = spacy.load("en_core_web_lg")
nlp.add_pipe("language_detector")

def clean(sentence):
    cleaned_text = list()
  text = sentence.split() #sentence corresponds to the content/sentence of a
cell in the column 'Name'; text then contains a list of terms within the
sentence
    for term in text:
        doc = nlp(term)
        if(doc._.language == 'en'):
            cleaned_text.append(doc.text)
    return " ".join(cleaned_text)
```

Listing 12 Removal non-English words

Project specific terms

```
#removal of words with digits
def remove_words_with_digits(text):
    return re.sub(r'\b\w*\d\w*\b', '', text)

df['Name'] = df['Name'].astype(str).apply(remove_words_with_digits)

#removal of words that indicate a direction
directions = ["north", "south", "east", "west", "northeast", "northwest",
"southeast", "southwest", "middle"]

pattern1 = r'\b(?:' + '|'.join(directions) + r')\b'

def remove_directions(text):
    return re.sub(pattern1, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_directions)

#removal of words that indicate a compartment of the project
components = [ "part", "section", "component", "element", "segment", "level",
"axis", "phase", "sub"]

pattern2 = r'\b(?:' + '|'.join(components) + r')\b'

#removal of words that indicate a count
```

```
def remove_components(text):
    return re.sub(pattern2, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_components)

pattern3 =
r'\b(?:first|second|third|fourth|fifth|sixth|seventh|eighth|ninth|tenth)\b'

def remove_pattern3(text):
    return re.sub(pattern3, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_directions)
```

Listing 13 Categories of removal of project specific terms

```
#removal of residual abbreviations
abbriviations = "NV", "LS", "SG", "A", "B", "MS", "TC", "HS", "WT", "HD", "C",
"D", "E", "RP", "LD", "NV", "NW", "NE", "KWW", "W", "SN", "NAP", "UWC", "HWA",
"GEWI", "m", "ID", "DP", "NS", "km", "tbv", "HRL", "HRR", "NSA", "KZNPD",
"AMSWG", "DDI", "ALAP", "HWN", "GWW", "BLVC", "H", "incl", "KZN", "UPS",
"KKW", "L", "j", "VERG", "SHO", "ZOAB", "F", "WLS", "SCVCAUO", "PDR", "TBGN",
"ab", "VV", "RFP", "ITP", "GK", "VERG", "RWS", "sh", "AMS"]

pattern3 = r'\b(?:' + '|'.join(abbriviations) + r')\b'

def remove_abbriviations(text):
    return re.sub(pattern3, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_abbriviations)

#removal of residual words
other_words = "end", "days", "expired", "ramp", "closure", "Type",
"deprecated", "noniFAT"]

pattern4 = r'\b(?:' + '|'.join(other_words) + r')\b'

def remove_other_words(text):
    return re.sub(pattern4, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_other_words)
```

Listing 14 Removal residual abbreviations and word

Appendix . When looking at Figure 28, it shows the boxplot of the original data. This boxplot shows that the mean and median are situated between 0,30 and 0,40 on the similarity score. Furthermore, it can be seen that the distance between the third quartile line and the maximum line is quite large. This means that not a lot of clusters (only 25% of the total made clusters) have a similarity score above the 0,553. It can be concluded that the clusters made with the original data are not that similar.

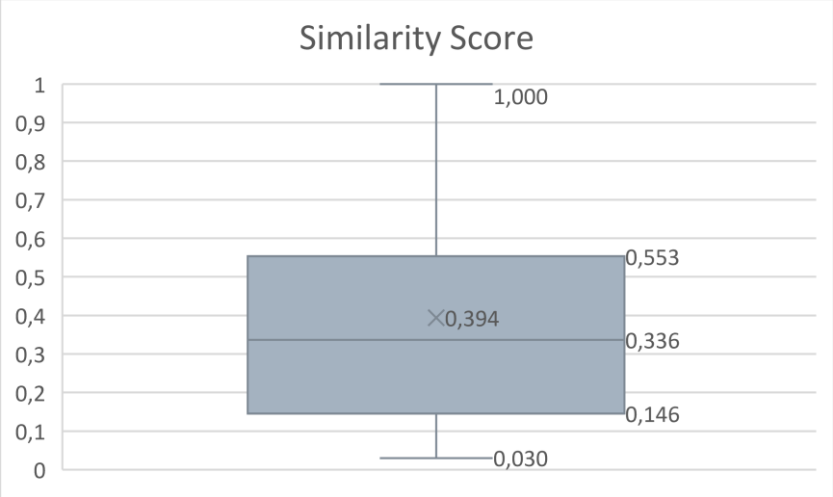


Figure 28 Boxplot of similarity score of original data

Examining Figure 29 reveals the distribution of clusters according to both the number of activities within a cluster and the corresponding similarity scores. As previously indicated in Figure 28, the majority of the clusters have a similarity score below the 0.5. Furthermore, Figure 29 illustrates a trend where clusters with a higher number of activities tend to have lower similarity scores, with a few exceptions represented by outliers.+

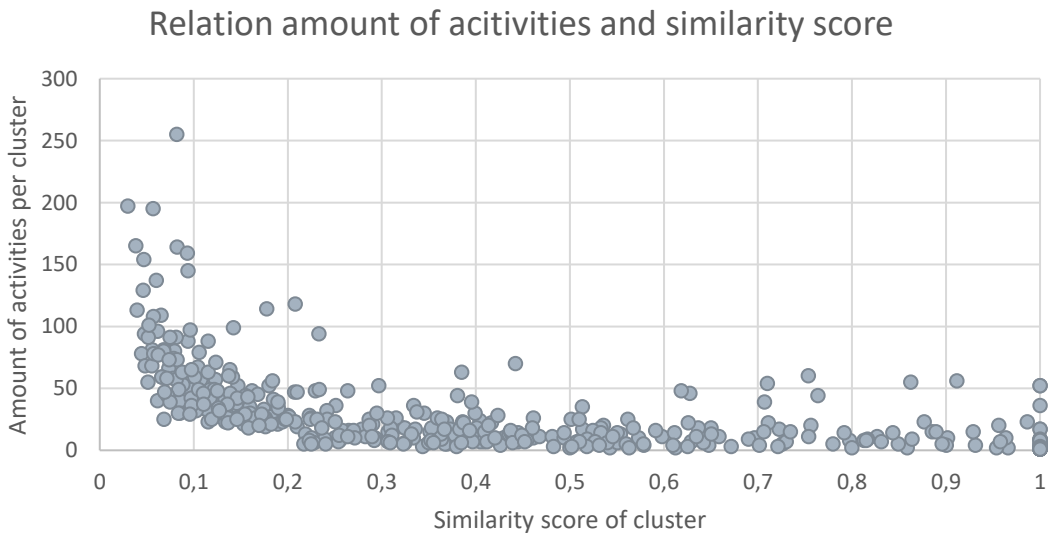
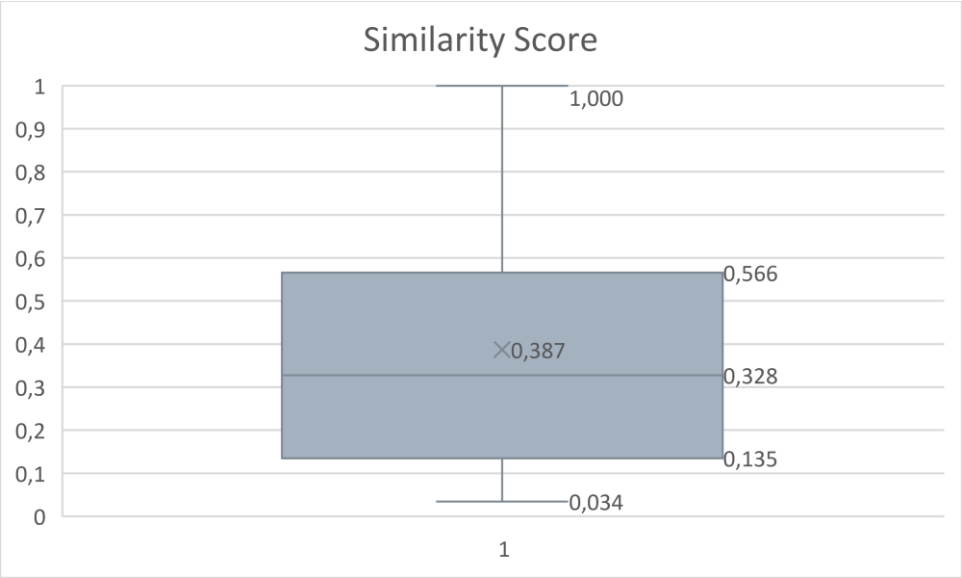
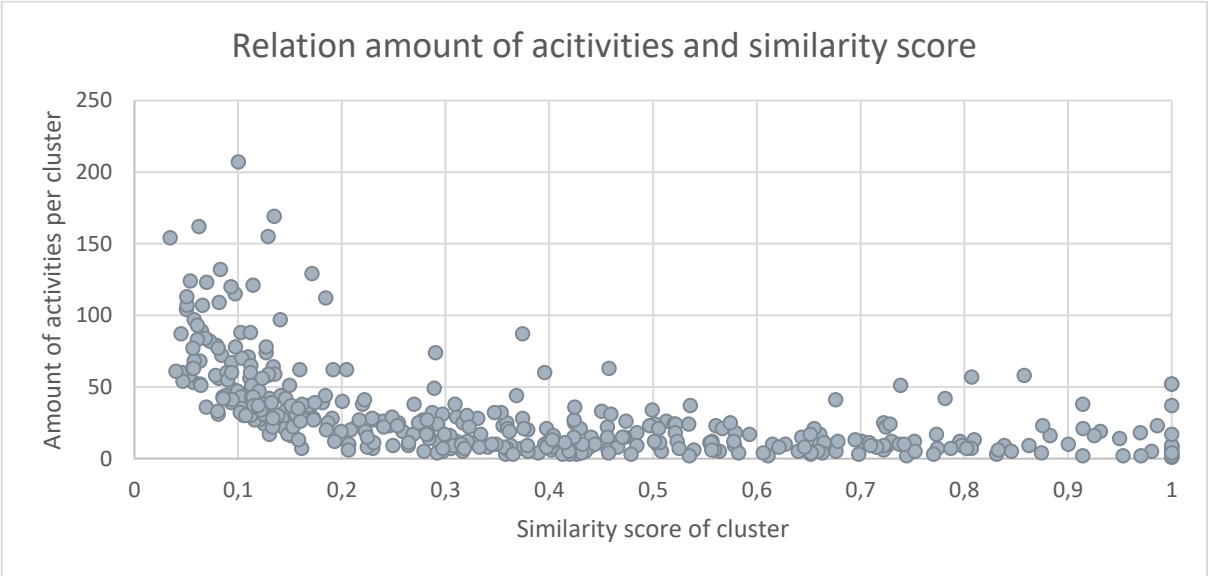


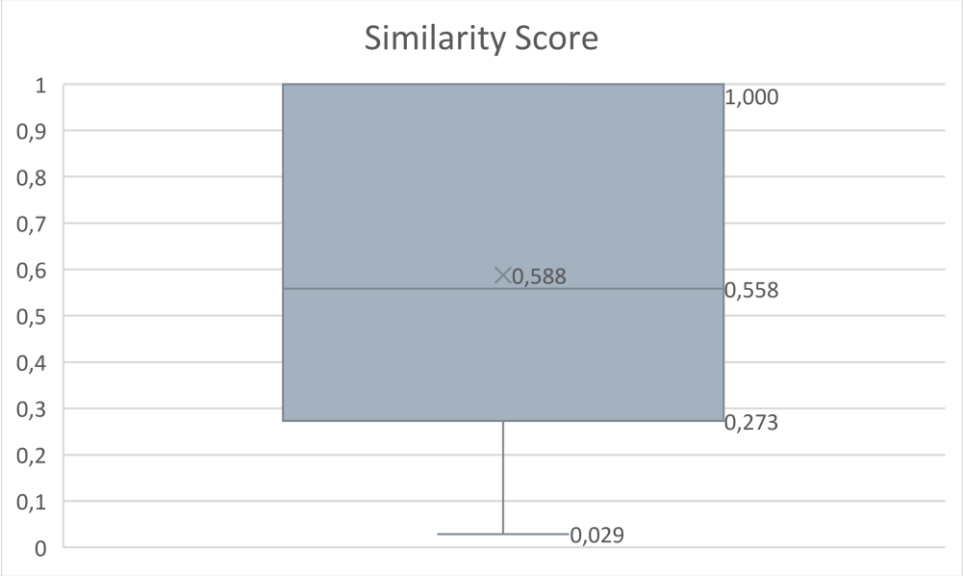
Figure 29 Distribution of activities

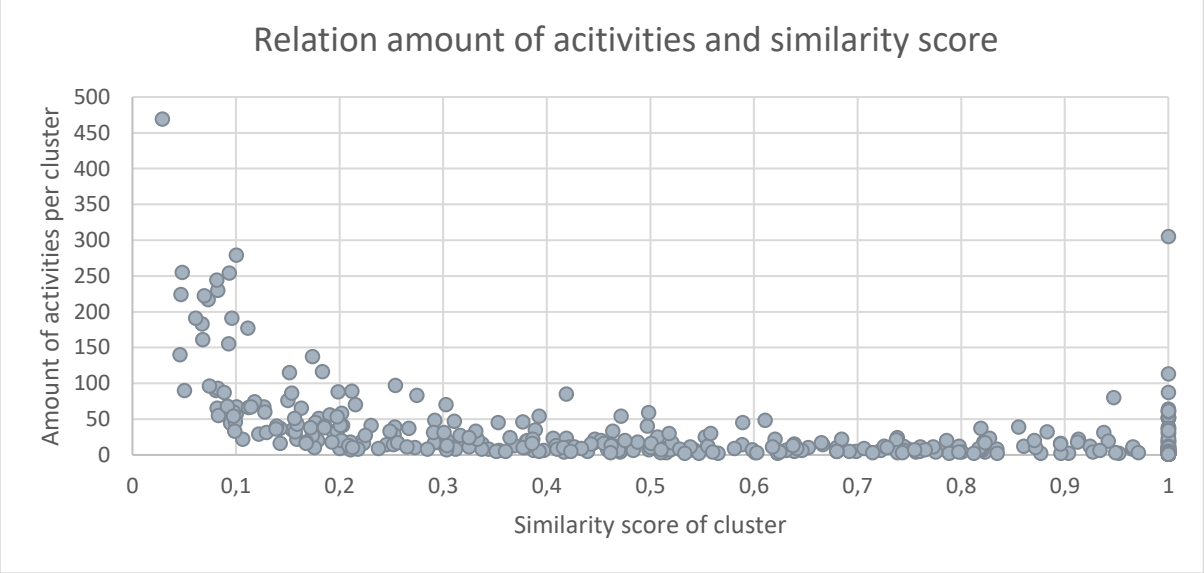
Word stemming


Similarity Score
1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
1,000
0,566
0,387
0,328
0,135
0,034
1

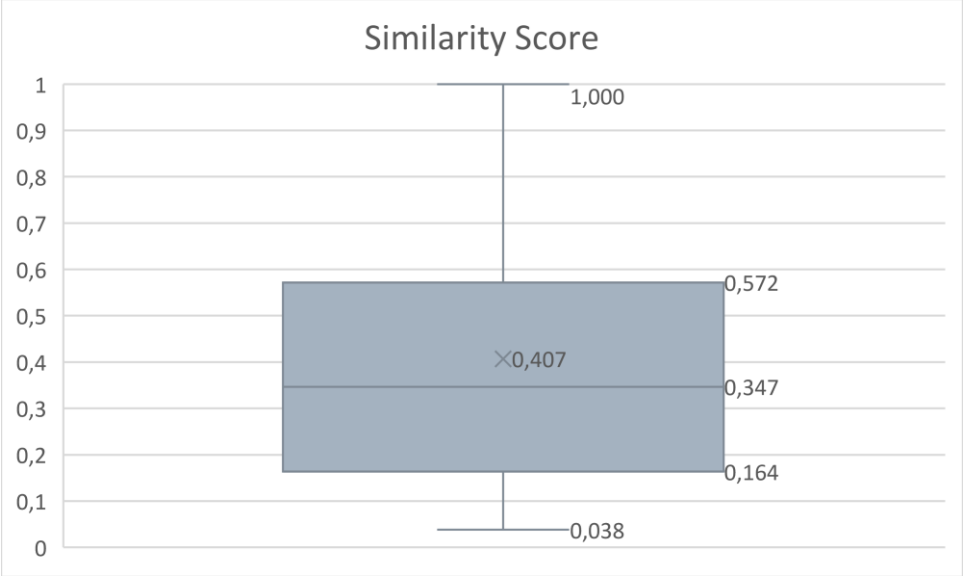

Relation amount of acitivities and similarity score
Amount of activities per cluster
250
200
150
100
50
0
0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
0,8
0,9
1
Similarity score of cluster

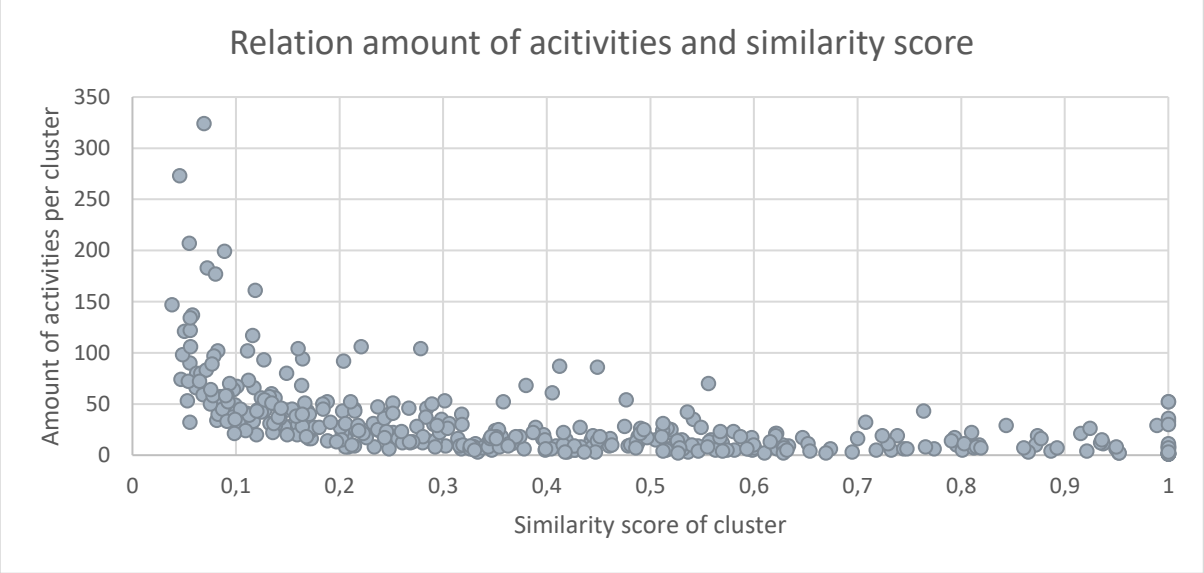
Part of Speech



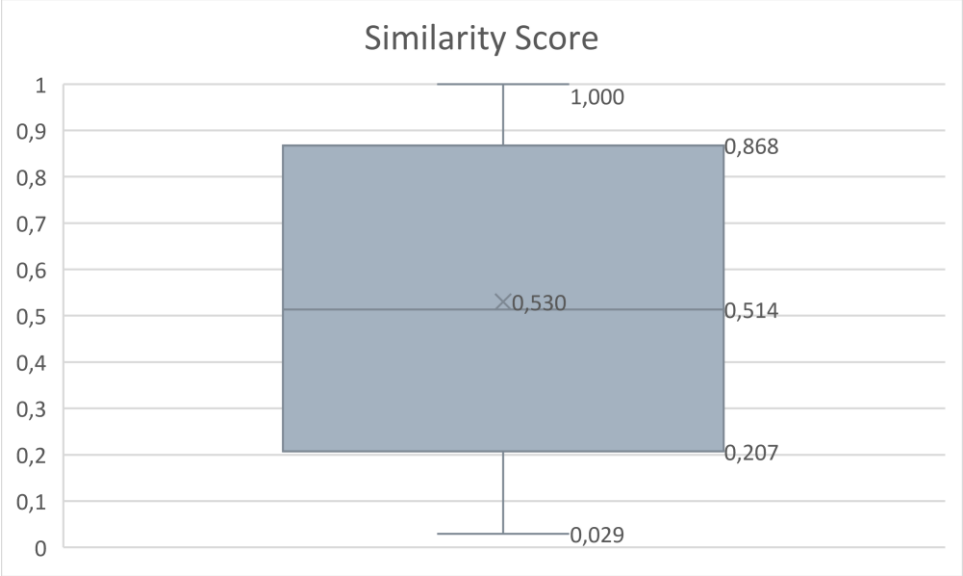


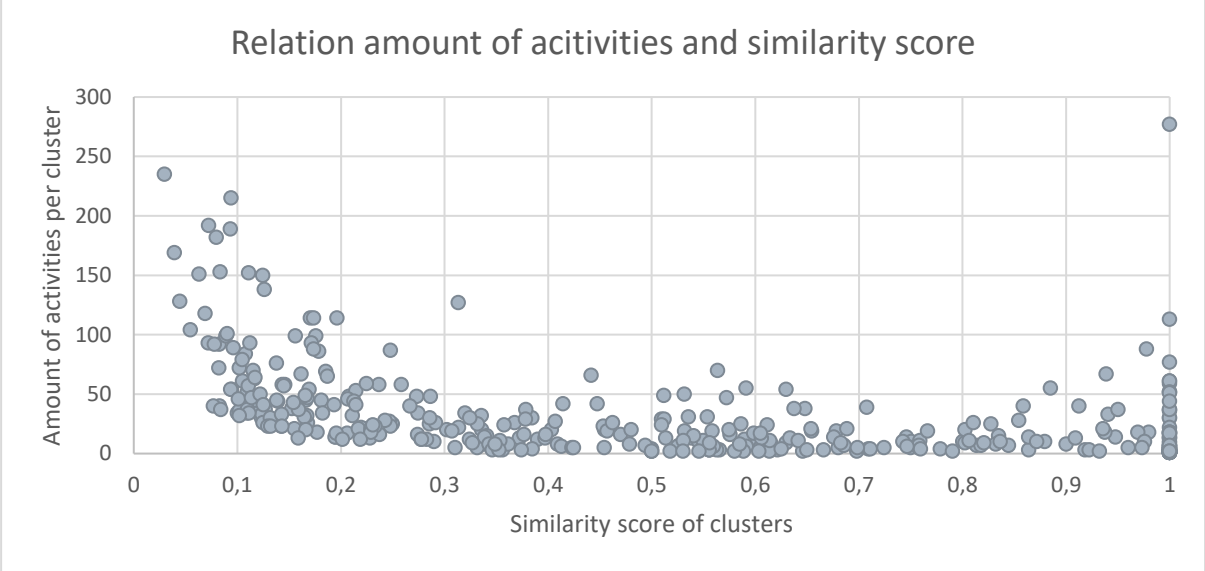
Punctuation marks



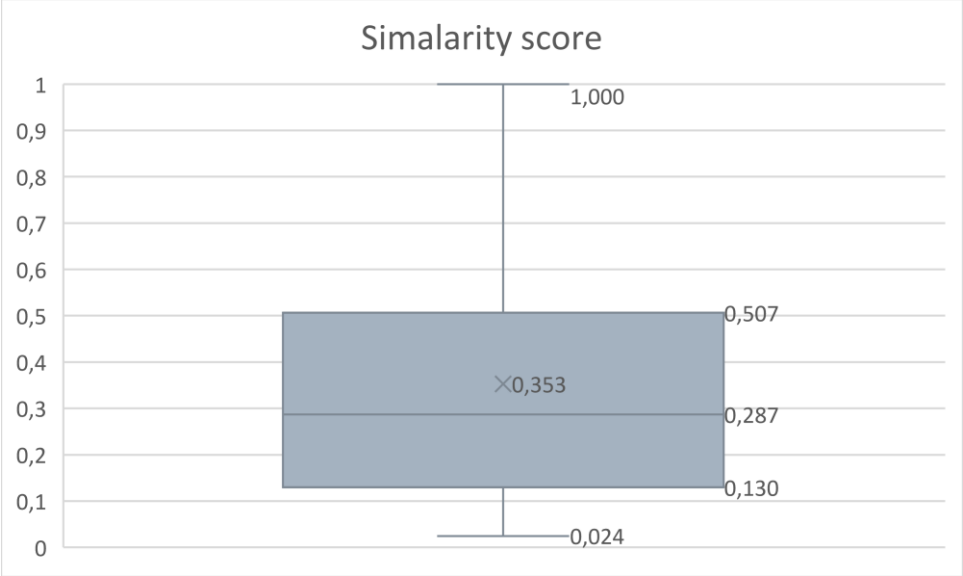


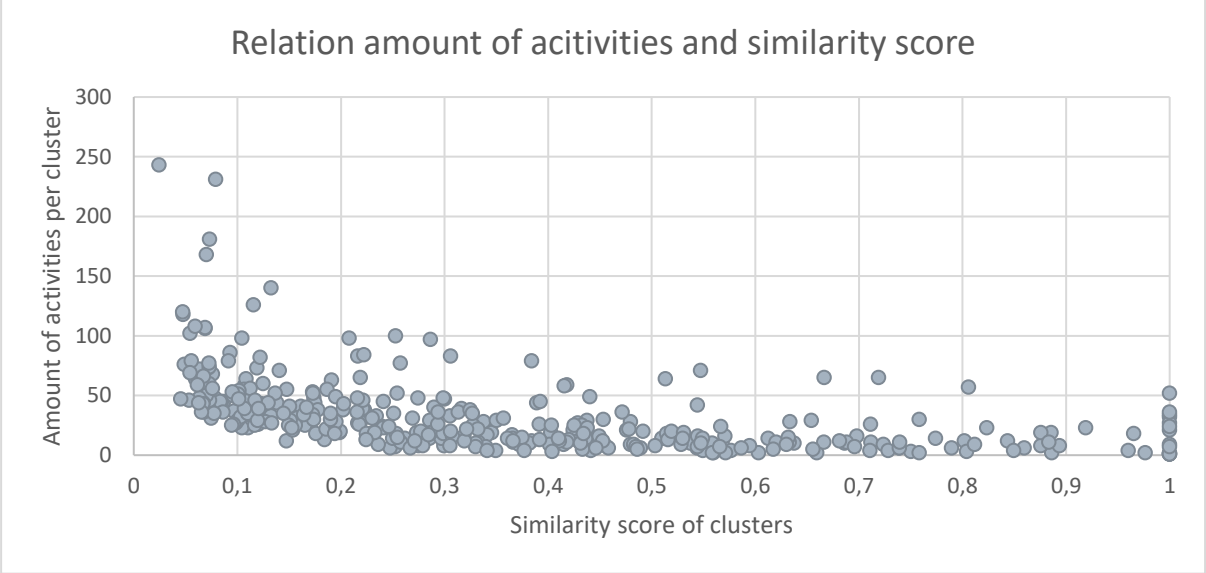
Project specific terms and abbreviations



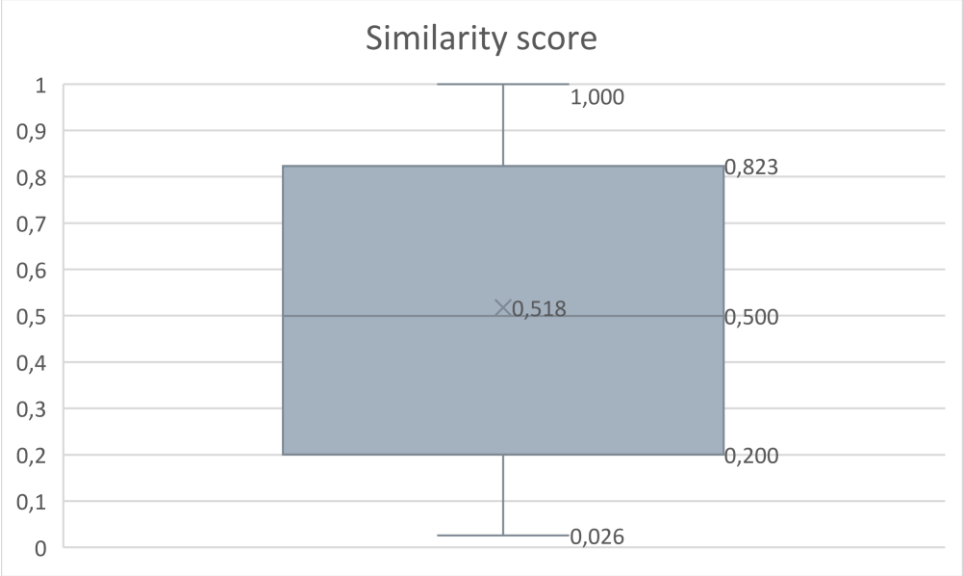


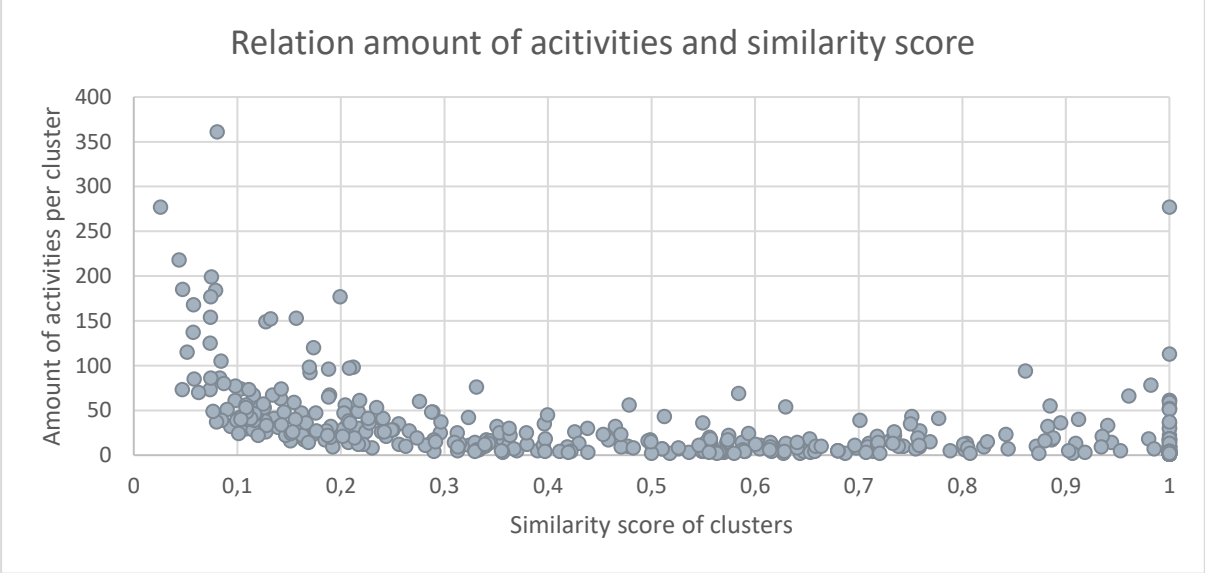
Removal Dutch words



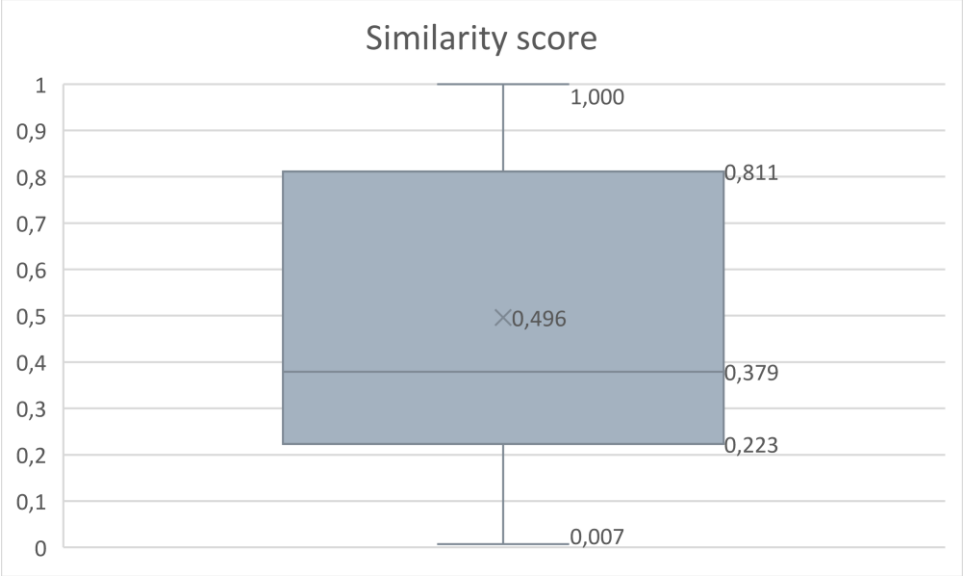


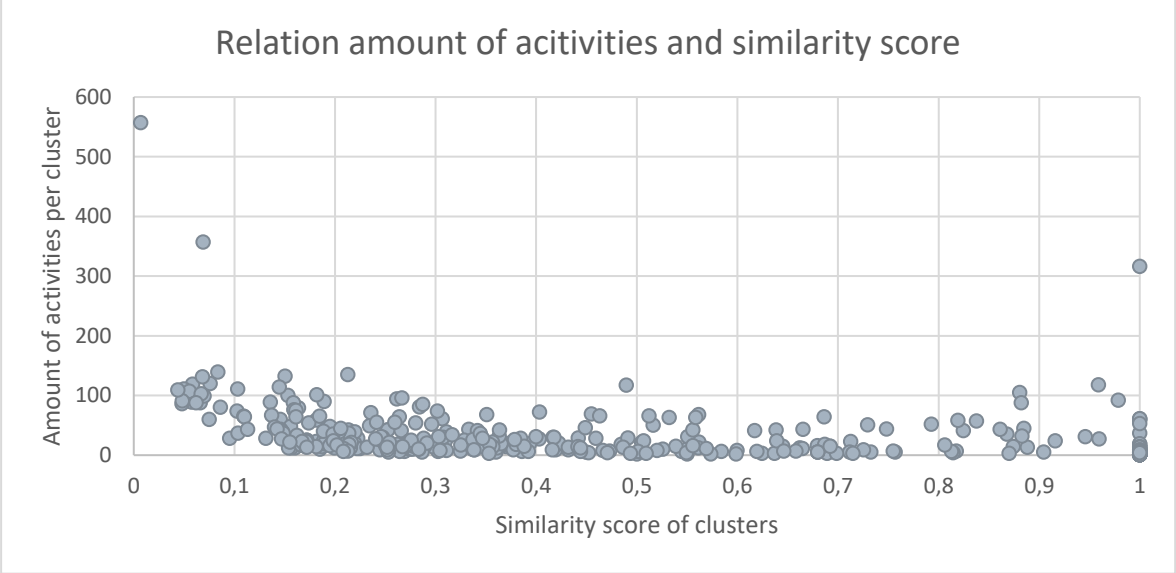
Only preparation with positive outcome





All


Similarity score
1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
1,000
0,811
0,496
0,379
0,223
0,007


Relation amount of acitivities and similarity score
Amount of activities per cluster
600
500
400
300
200
100
0
0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
0,8
0,9
1
Similarity score of clusters

	Mean	Median	Min	Q1	Q3	Max
Original data	0,394	0,336	0,030	0,146	0,553	1
Stemmed data	0,387	0,328	0,034	0,135	0,566	1
Part of Speech defined data	0,588	0,558	0,029	0,273	1	1
Punctuation cleaned data	0,407	0,347	0,038	0,164	0,572	1
Project specific terms	0,530	0,514	0,029	0,207	0,868	1
Removal Dutch words	0,353	0,287	0,024	0,130	0,507	1
Only positive improvement adaptions	0,518	0,500	0,026	0,200	0,823	1
All adaptions (sequence: project specific words – brackets+punctuation – PoS – language)	0,496	0,379	0,007	0,223	0,811	1
All adaptions (sequence: PoS – brackets – language – project specific words – stemmer)	0,450	0,373	0,042	0,206	0,645	1

Comparison scoring methods

The silhouette score was chosen as the primary evaluation measure because of its ability to provide a complete view of the clustering quality. This measure takes into account not just semantic similarity but also the clustering's overall structure. This assures that the resulting clusters are defined in an optimal way. This scoring method is used, because of the extensive data set that is used in this research. While semantic similarity is essential, the formation of correctly differentiated clusters is as important. However, the similarity score is not completely ignored. This scoring method gives valuable insight into the distribution of the clusters when comparing the size of clusters to the similarity score. Therefore, this information is present in **Error! Reference source not found.** as well a s in Appendix .

	Silhouette score (estimation)	**Cosine similarity**	
		Mean	**Median**
Original data	0,435	0,394	0,336
Stemmed data Lemmatization data	0,435	0,387	0,328
Part of Speech defined data	0,560	0,588	0,558
Punctuation cleaned data	0,455	0,407	0,347
Project specific terms	0,550	0,530	0,514
Removal Dutch words	0,455	0,353	0,287
Only positive improvement adaptions (PoS, punctuation, project specific terms)	0,540	0,518	0,500
All adaptions (sequence: project specific words - brackets+punctuation - PoS – language)	0,580	0,496	0,379
All adaptions (sequence: PoS - brackets - language – project specific words - stemmer)	0,545	0,45	0,373

Appendix H: Code for generating the name of the labels

Clustering algorithm based on most prevalent words in each cluster:

```
df['Name'].fillna('', inplace=True)

grouped_df = df.groupby('Cluster')['Name'].apply(lambda x: '
'.join(x)).reset_index()
r = Rake()

result_df = pd.DataFrame(columns=['Cluster', 'Score', 'Phrase'])

# Extract keywords, so stopwords are not used
for index, row in grouped_df.iterrows():
    cluster = row['Cluster']
    text = row['Name']

    r.extract_keywords_from_text(text)
    ranked_phrases_with_scores = r.get_ranked_phrases_with_scores()

    temp_df = pd.DataFrame({'Cluster': [cluster] *
len(ranked_phrases_with_scores),
                            'Score': [score for score, _ in
ranked_phrases_with_scores],
                            'Phrase': [phrase for _, phrase in
ranked_phrases_with_scores]})
    result_df = pd.concat([result_df, temp_df], ignore_index=True)
```

Clustering based on most common name:

```
# Group the data by 'Cluster' and calculate the most common activity for each
cluster
cluster_Name = data.groupby('Cluster')['Name'].agg(lambda x:
x.value_counts().idxmax())

results_df = pd.DataFrame(cluster_Name).reset_index()
results_df.columns = ['Cluster', 'Most_Common_Activity']
```

Appendix I: Outcome labels with similarity score

Cluster	Label	Similarity
0	Application formworks	25%
1	Apply sand	93%
2	work floor	100%
3	Apply RIWP process execution	97%
4	Moot floor Formwork bottomlayer	100%
5	Installation prefab beam	95%
6	Apply sand step	80%
7	Making workspace	60%
8	Moot floor Making workspace	86%
9	Moot floor Concrete hardening	100%
10	Apply traffic	79%
11	Apply base Intermediate traffic loop onramp traffic	50%
12	installation pc formwork	53%
13	water way	14%
14	Concrete Works pile cap	36%
15	Moot floor Installation Demus wall	100%
16	Backfilling Abutment	28%
17	Auxiliary bridge side Apply Support Construction beam	89%
18	Apply	99%
19	Apply road foundation	86%
20	Apply sand mtr	80%
21	installation Pillar	27%
22	Moot Applying wall	93%
23	Apply preloading	50%
24	Welding sheetpiles	43%
25	Moot floor Reinforcement cleaning	100%
26	concrete	84%
27	Apply RIWP process	100%
28	Apply filter drainage	91%
29	Delivery Time Engineering Deliver Apply Rainwater sewerage	33%
30	Ground work work floor Abutment	60%
31	Setting plan traffic measure	64%
32	Prepare input scoperequirements Proc	34%
33	Review Concrete wall	33%
34	Apply beam Casting	100%
35	Install working platform installation pc	21%
36	Cleaning surface placing scaffolding	51%
37	Execution Plan Release Execution	31%
38	Applying wall	81%
39	Install	19%
40	Moot Wall execution	100%
41	Apply Buffer sand week	94%
42	Hardening tensioning wall	46%
43	Moot floor Welding steelplate sheetpileconcrete floor	99%

44	Prepare apply beam Casting	100%
45	TIS Concept Work Prep UO Batch	17%
46	Period Deliver Apply Sealing profile	60%
47	Moot floor Support Formwork	100%
48	Approval TMP converting existing traffic stop weekend	10%
49	Welding Theads sheet pile Bottom Beam	81%
50	Auxiliary bridge side Apply Support Construction Basement	100%
51	Reinforcement cleaning	81%
52	Finish	32%
53	Concrete hardening	97%
54	Apply week	100%
55	Remove soil	21%
56	Workfloor apply beam Casting	100%
57	Realizing workfloor sand	33%
58	Design UO Batch side waterside Foundation pile	67%
59	Apply road marking	100%
60	strip deck side knevelen existing	13%
61	Make provision prefab column	11%
62	Moot ITSO Wall	49%
63	Apply sand OnOff Ramps	95%
64	Finishing pile	20%
65	Apply sand G Waterway	100%
66	Auxiliary bridge side Apply Support Construction Abutment beam	100%
67	Moot Wall Concrete hardening	100%
68	side steelwork drilling reinforcement coupler	59%
69	Sheet piling compartment screen	21%
70	Delivery Time Concrete Works Formwork Concrete Application	44%
71	Installation sheet pile	77%
72	Apply deck	100%
73	Delivery Time Deliver Apply Grounding 131warding reinforcement plate cadweld	100%
74	Auxiliary bridge side Apply Support Construction outer beam	100%
75	ground retaining wall front wall	31%
76	Apply base layer intermediate area milling connection	100%
77	Abutment concrete	30%
78	Internal preparation Obtaining Building Permit Construct Soil Barrier	22%
79	Apply sand ZO	67%
80	Install sewage system Casing pipe apply road foundation traffic	71%
81	Apply traffic measure	100%
82	Pavement foundation construction Deepcut	11%
83	Apply base Intermediate traffic loop traffic	73%
84	Transport muck excavation comp	18%
85	placing transition plate Abutment	21%
86	removing formwork finishing concrete	70%
87	side formwork plywood	21%
88	Apply preloading layer layerweek layer overheight	67%
89	Moot cleaning comp cleaning pile	78%

90	Moot preparation	68%
91	Concrete side Pouring concrete deck POUR DATE	38%
92	Delivery Time Execution Foundations	39%
93	Braiding formwork slab	40%
94	Formwork bottomlayer	65%
95	Draft release report	25%
96	Reinforcement Bottom Beam	71%
97	Period Drilling work Apply promatec roof	100%
98	Apply Reinforced Soil G Waterway	43%
99	Apply support scaffolding formwork Deck	100%
100	Casting beam Casting curing	55%
101	Review pile	22%
102	installation pc	36%
103	Moot Installing connector dish	68%
104	Formwork beam Casting	92%
105	Finishing Roadworks step Traffic	50%
106	Moot Hardening tensioning wall	100%
107	Moot Draining compartment	62%
108	Apply sand Fugro needed	100%
109	Apply sand week	100%
110	Deck side drilling coupler roughening	13%
111	Support Formwork	98%
112	Delivery Time EPS	18%
113	Placing beam SPAN	56%
114	Apply sand Reinforced Soil	100%
115	Concrete work floor wall Abutment	100%
116	Apply pavement	50%
117	drainage	22%
118	Auxiliary bridge side Apply Support Construction Abutment outer beam	100%
119	Realize bearing beam base Abutment	50%
120	Formwork Concrete Curing Period Bottom Beam	100%
121	Finishing Roadworks barrier marking signage traffic	11%
122	Applying source drainage Pillar	10%
123	report bridge	6%
124	Auxiliary bridge side Apply Support Construction CLOSE	100%
125	Delivery time	35%
126	Apply base Intermediate	71%
127	Apply reinforcement Casting	100%
128	Finishing pile Abutment	58%
129	Placing support beam	56%
130	Compression formwork	68%
131	Assessment award	9%
132	Release report review	43%
133	Apply base Intermediate connection traffic	73%
134	Stabilized sand	80%
135	Placing beam beam	39%

136	Delivery Time Deliver Apply Remove Drainge excavation Dewatering	100%
137	Traffic loop	60%
138	Moot floor Formwork wgu bottomlayer	94%
139	Moot Underwater concrete curing time	89%
140	Pavement foundation construction	40%
141	Casting beam Casting curing inspection	50%
142	Remove support construction deck sewage pump cellar KKWeg	17%
143	Remove existing structure	25%
144	Front wall wing wall Abutment	40%
145	Remove preloading	51%
146	Execution Plan Draft	36%
147	Fishing water way	11%
148	Apply filling	100%
149	Remove crossbeam beam support	67%
150	Delivery Time Engineering Deliver Apply Remove Temporary shipping collision barrier	100%
151	Bushing existing abutment	33%
152	Installing	60%
153	Delivery Time Deliver Apply Special Anchors ITSO K Anchors	100%
154	Install bearing block Abutment	14%
155	Period Deliver Apply Remove Drainge excavation Dewatering	75%
156	release party	25%
157	Draft Deliver Apply front wall	100%
158	Technical review UO Batch	20%
159	Secure beam	28%
160	Apply sand week OWN	100%
161	Monitoringsplan execution	38%
162	Moot Reinforcement cleaning	67%
163	Hardening tensioning phasing	19%
164	Testing floor pile	93%
165	Apply sand SOIL	100%
166	Remove cleanup verge traffic	12%
167	joint	12%
168	Connection abutment	55%
169	Installation beam Moot	100%
170	Apply signes	93%
171	Finishing floor wall	75%
172	Send Verifcation Validation plan concept review	14%
173	Delivery Time Groundworks Delivery apply	50%
174	Concrete work floor	60%
175	Applying road foundation	76%
176	Moot Installation prefab beam Compartment Moot	100%
177	Abutment beam formwork including wing Abutment	89%
178	Apply adding lane rearranging existing road weekend	33%
179	Abutment beam soil Abutment	80%
180	Apply filling location existing	100%
181	Apply bicycle path	50%

182	Realize foundation slab Abutment	48%
183	Installation support beam	50%
184	Rebar formworks concrete connection beam	93%
185	Install sewage system gutter	14%
186	Finalize Concrete Testing	30%
187	Apply rail	100%
188	Moot Installing phasing wall	57%
189	Auxiliary bridge side Apply Support Construction Basement AFTER Basementwall	100%
190	Abutment beam Burning pile pc Abutment	31%
191	Processing review UO Batch	28%
192	Casting Finishing pile Pillar	19%
193	Submit Factsheets Municipality	21%
194	Install sewage system Casing pipe apply road foundation temp traffic	44%
195	Removal shoring Abutment	17%
196	side support beam prestressing	77%
197	Abutment beam formwork finishing concrete Abutment	100%
198	Moot Concrete hardening	100%
199	Moot Installing gravel	47%
200	Beams	8%
201	Dry excavation sand	100%
202	Step Roads work	22%
203	Period Deliver Apply Special Anchors ITSO K Anchors	80%
204	Moot Installing wall	100%
205	Assembling prefab front wall promatec panel	66%
206	soil deck	57%
207	Work platform week	38%
208	Processing FastTrack Review UO Civil Structures Concrete Batch Columns	27%
209	Apply intermediate	100%
210	Finishing deck	16%
211	Abutment beam connecting reinforcement Abutment	47%
212	FastTrack Review UO Civil Structures Concrete Batch Columns	26%
213	Submit Detailed building safetyplan ODNZKG execution	33%
214	Installing element	27%
215	Side formworks	39%
216	Apply Remaining Layers Preloading	33%
217	Place Jacking Support existing	25%
218	Apply working platform	50%
219	elevating sand mtr offramp	80%
220	installation Abutment	75%
221	Period Deliver Apply Grounding 134warding reinforcement plate cadweld	100%
222	time Tbeams	9%
223	Moot Leveling	48%
224	Set Review Package Flow UO Batch	15%
225	Moot Installing foundation pile noise barrier	54%
226	Abutment Braid Reinforcement	23%
227	Execution	63%

228	place reinforcement pile	33%
229	Installation beam Compartment Moot	83%
230	Fill soil area	21%
231	Finishing pile Pillar	18%
232	Apply connect existing exit	50%
233	Design Road Relocations Works	12%
234	Apply foundation construction	100%
235	Foundation slab placement	30%
236	Submit Factsheets Municipality execution	63%
237	Draft Deliver Apply Remove Drainge excavation Dewatering	43%
238	Moot floor pile	100%
239	placing formwork	64%
240	Process Review Finalize	29%
241	Hardening time	45%
242	Moot Muck Excavation	52%
243	Apply location existing	100%
244	Placing beam piece beam	15%
245	road including settlement compensation	13%
246	Concrete side Hardening concrete	67%
247	Install door	5%
248	Bearing Block replacement existing	11%
249	Prestressing concrete stressing head	13%
250	Apply CONNECTION ARCH V	50%
251	Wet knot support connecting reinforcement beam	29%
252	Installation testing equipment	15%
253	cleaning comp cleaning pile	41%
254	Placing beam	27%
255	Connection support point	25%
256	strip deck side making formwork waiting time settlement	33%
257	Auxiliary bridge side Apply Support Construction CLOSE week	50%
258	Columns Apply formwork column casing	67%
259	Making starting point reinforcement work Bottom Beam	59%
260	Road construction Foundation Drainage	33%
261	Apply sand gravel	100%
262	Placing beam Field	55%
263	Apply road foundation EPS	100%
264	DO execution note Hand	10%
265	side	33%
266	Runon slab placement	14%
267	Placing support mounting beam beam	77%
268	Cut Investigate safetyzones apply traffic measure Municipality	50%
269	Suppliers bid time Proc	34%
270	Moot Installation Demus wall	100%
271	Installation sheet pile weekend week Abutment	100%
272	Concrete side Hardening concrete deck connecting	100%
273	Installing wall	59%

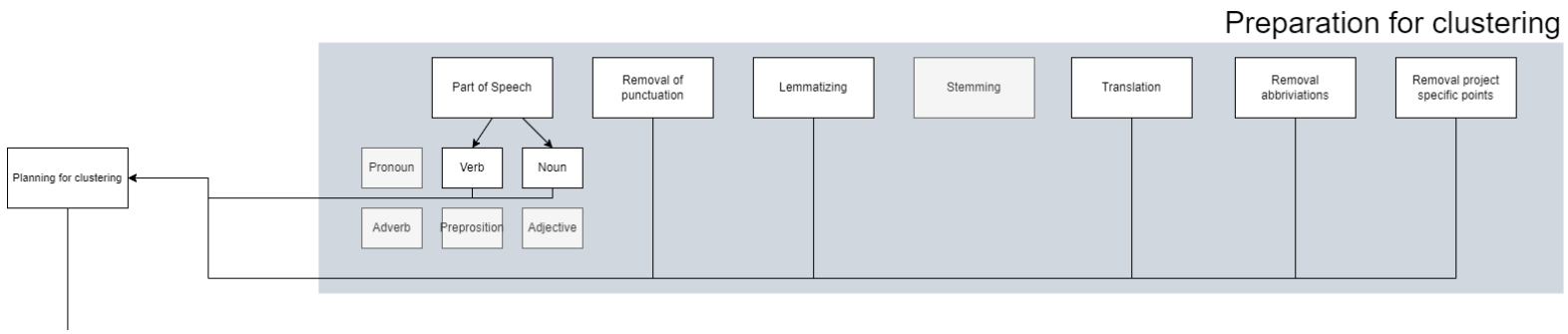
274	Casting wall Abutment	100%
275	Floor Moot	100%
276	Apply sand layer finish	100%
277	Apply Reinforced Soil G	29%
278	Proxy Waiting time	10%
279	Workfloor beam Casting	38%
280	Installation deck Week	100%
281	Install traffic loop Traffic	67%
282	Apply road foundation sand foundation	100%
283	baseplate support block support block	47%
284	Formwork Concrete Casting Casting curing	40%
285	Abutment beam slab Abutment	50%
286	Delivery Time Deliver Apply Architectural finish	33%
287	Pavement base construction intermediate	47%
288	Period	14%
289	Fill existing water way soil area	48%
290	Moot Welding floor	100%
291	Remove portal VDC	18%
292	Placing barrier finishing deck	38%
293	Apply deck sewage pump cellar KKWeg	29%
294	Apply sand eventueel	50%
295	Freeze period change week step Traffic	25%
296	Install work floor	50%
297	Auxiliary bridge side Apply Support Construction Abutment	100%
298	Draft Deliver Apply Grounding 136warding reinforcement plate cadweld	100%
299	Executing Sheet pile construction pit	53%
300	Delivery Time Beams	50%
301	Compartment Lava execution	42%
302	Auxiliary bridge side Cut pile weld base plate beam Abutment	44%
303	top traffic	40%
304	existing foundation slab	12%
305	FastTrack Review UO Civil Structures Concrete Batch Foundation pile	63%
306	Period Groundworks Delivery apply	100%
307	Welding steel ring pile Burning pile pc	32%
308	time start work	76%
309	Placing Fundex pile	33%
310	Delivery Time Drilling work Apply promatec roof	100%
311	Casting slab Abutment	27%
312	Ground improvement work floor Abutment	44%
313	Casting Cross Beam Pillar	31%
314	Apply base Intermediate connection temp Sarch traffic	50%
315	Abutment beam concrete Abutment	100%
316	Apply roadmarking	86%
317	Bearing Block replacement existing Abutment	17%
318	Abutment Install form work	26%
319	Casting space prefab wall inspection	70%

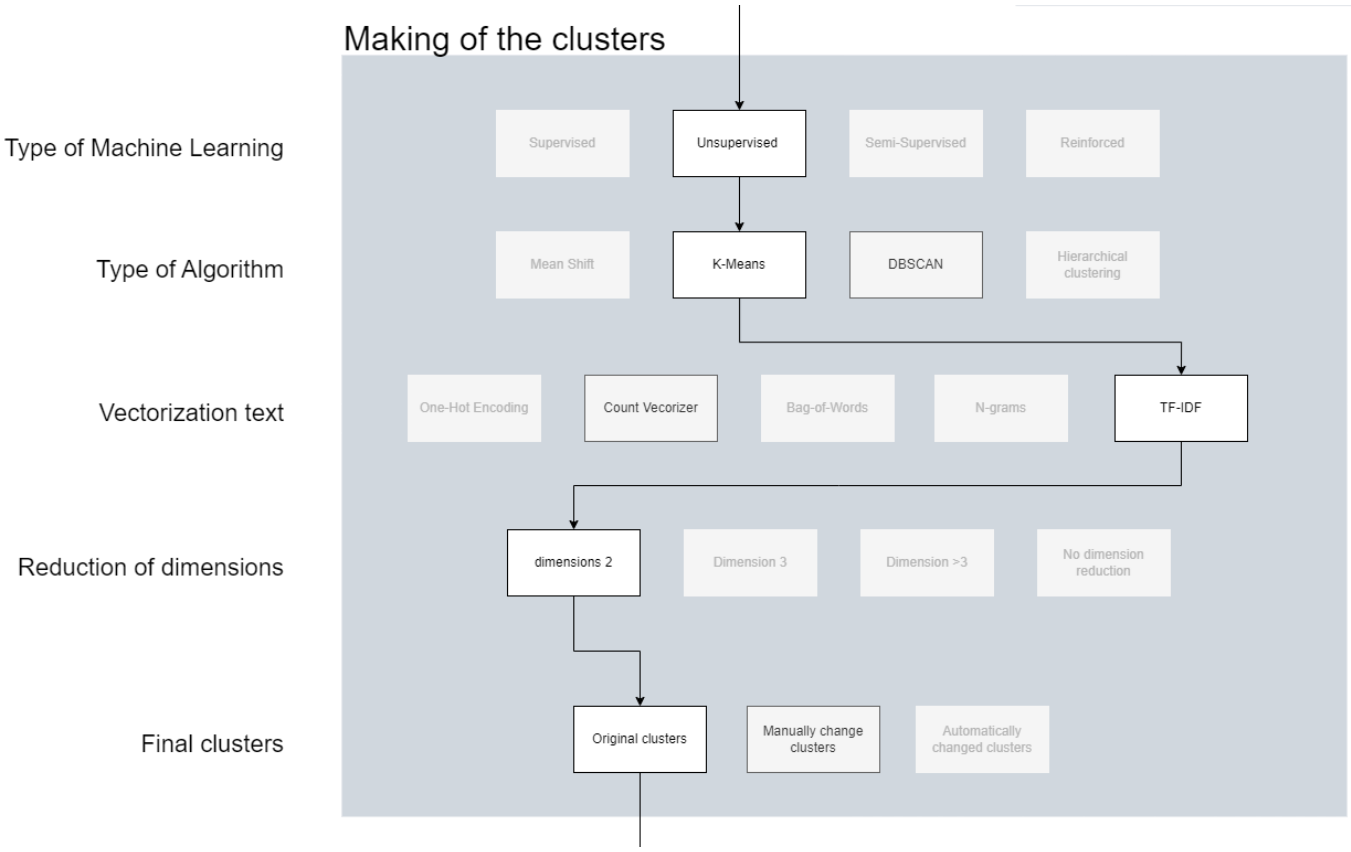
320	Apply remaining prefab deck Basement situation bridge flap	50%
321	Wall execution	100%
322	Design UO Batch side waterside Columns	25%
323	Abutment	41%
324	Concrete work cross beam	71%
325	Burning sheet pile beam Casting	74%
326	Moot Dry excavation peat	33%
327	Apply onramp traffic	75%
328	Installation sheet pile abutment	33%
329	Process Review Finalize delivery package barrier	12%
330	Factsheet Road Works Make	12%
331	Delivery Time Deliver Apply Sealing profile	75%
332	Kick Off delivery package	31%
333	Placing pillar	25%
334	Install Final Soil Barrier pile	40%
335	scouring border	8%
336	Apply sewage pump cellar KKWeg	67%
337	Install LIANDER	12%
338	Investigate safetyzones apply traffic measure Municipality	100%
339	Process Review Finalize delivery package Concrete floor	67%
340	Processing FastTrack Review UO Civil Structures Concrete Batch Foundation pile	50%
341	Abutment Install facility	32%
342	Installation sheetpiles	19%
343	Apply plant eastbank Southside exemption start def work	100%
344	Apply base Intermediate traffic loop	50%
345	wall predrilling accordance	32%
346	Abutment Close Formwork Kist	44%
347	Apply deck situation	100%
348	Period Engineering Deliver Apply Rainwater sewerage	33%
349	Applying strut waling	26%
350	Moot Applying strut waling	64%
351	Produce delivery package document	35%
352	side formwork plywood Abutment	18%
353	building pit top foundation slab	20%
354	Finishing work	44%
355	Moot roof Rebar formworks concrete connection beam	100%
356	Verification Plan Draft Internal Review	9%
357	Delivery Time Engineering Deliver Apply Joints deck	100%
358	Internal preparation Obtaining Demolition Notice Demolish	6%
359	Install remaining EPS Abutment	22%
360	Delivery Time Concrete Testing	18%
361	Installation EPS Abutment	73%
362	period plan	21%
363	Placing concrete panel	45%
364	installation	91%
365	Realize foundation slab Pillar	15%

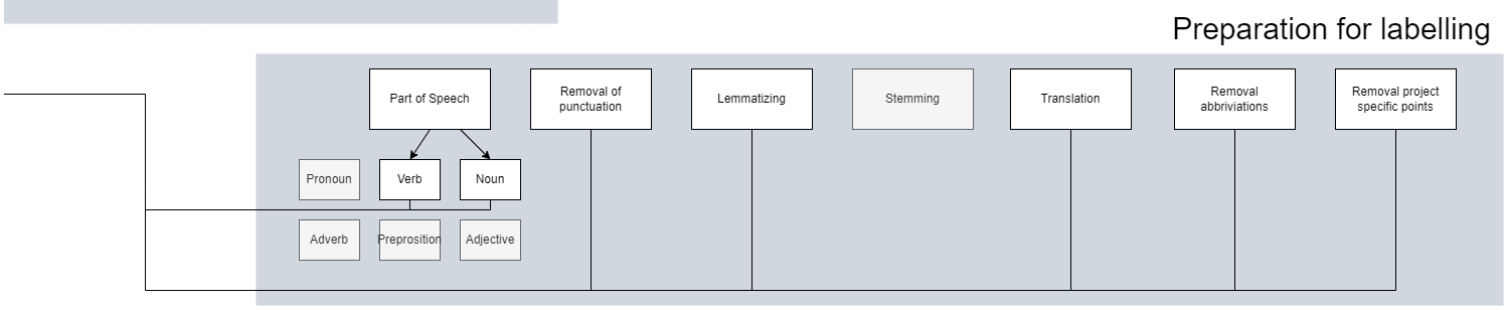
366	Burning pile pc	35%
367	Moot Installing	100%
368	G Waterway	11%
369	Abutment Hardening time	80%
370	Apply sand layer week period calendardays week	100%
371	Apply Buffer sand week reduced	50%
372	Backfilling building pit sand floor Basement	100%
373	time	40%
374	Apply floor sewage pump cellar KKWeg	100%
375	Process Review Finalize Delivery Package Preparatory work Traffic	50%
376	strip deck side hardening concrete	57%
377	Design UO Batch side landside Foundation pile	31%
378	period Concrete floor execution	100%
379	Apply layer	100%
380	Draft Drilling work Apply promatec roof	100%
381	Execution Plan Internal Review Process Comments	57%
382	Moot Cleaning pile total	100%
383	Execution concrete work	67%
384	Apply seepage screen sheetpiles	100%
385	Draft Deliver Apply Sealing profile	60%
386	Apply sewage system gutter	80%

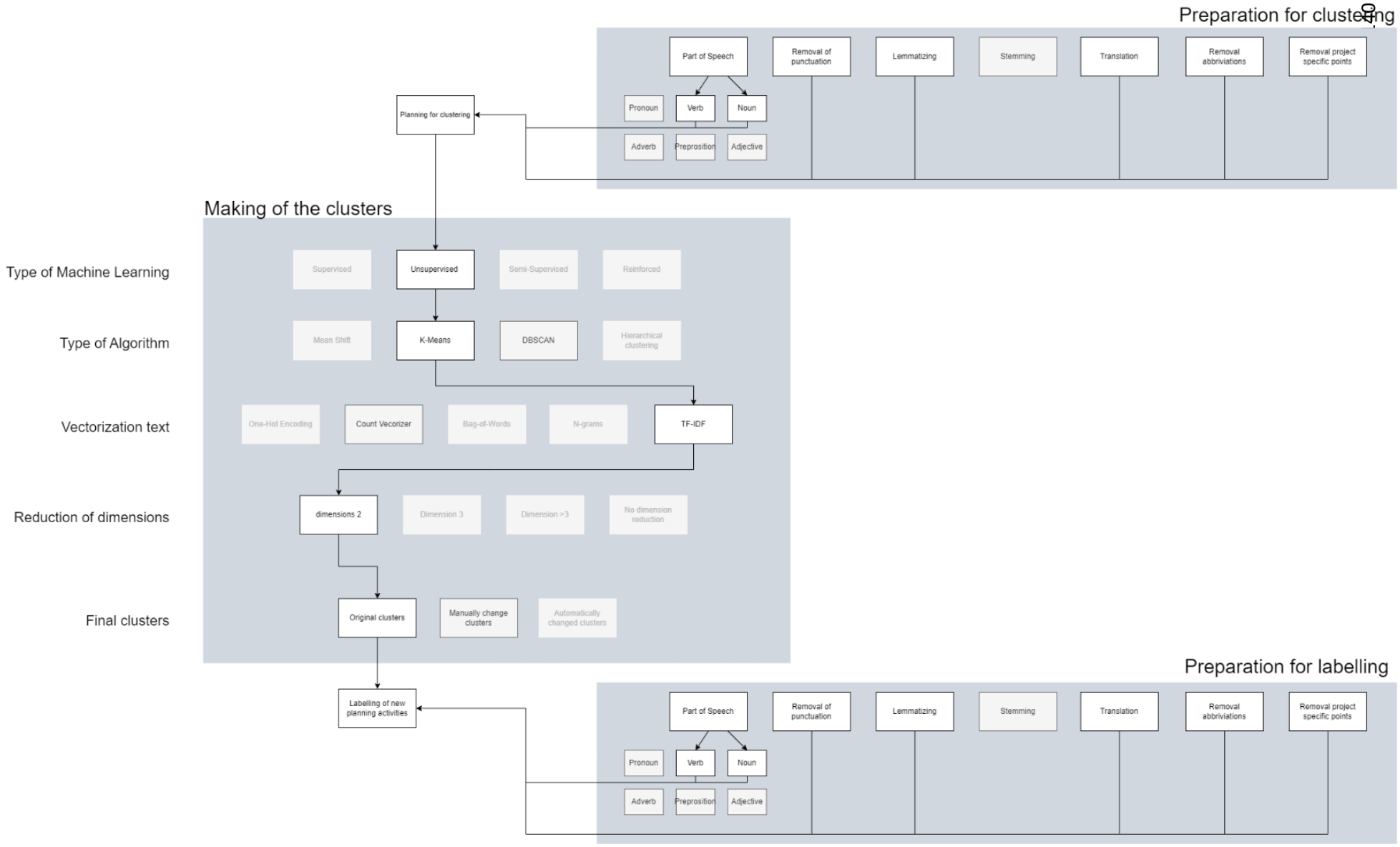
Appendix J: Entire path taken in Results chapter

In parts:








Preparation for clustering
Part of Speech
Removal of punctuation
Lemmatizing
Stemming
Translation
Removal abbriviations
Removal project specific points
Pronoun
Verb
Noun
Adverb
Preposition
Adjective
Planning for clustering
Making of the clusters
Type of Machine Learning
Supervised
Unsupervised
Semi-Supervised
Reinforced
Type of Algorithm
Mean Shift
K-Means
DBSCAN
Hierarchical clustering
Vectorization text
One-Hot Encoding
Count Vecorizer
Bag-of-Words
N-grams
TF-IDF
Reduction of dimensions
dimensions 2
Dimension 3
Dimension >3
No dimension reduction
Final clusters
Original clusters
Manually change clusters
Automatically changed clusters
Preparation for labelling
Part of Speech
Removal of punctuation
Lemmatizing
Stemming
Translation
Removal abbriviations
Removal project specific points
Pronoun
Verb
Noun
Adverb
Preposition
Adjective
Labelling of new planning activities

Appendix K: Adjusted code data preparation for labelling codes

```
#removal of residual abbreviations
abbriviations = "RKP", "KW", "AD", "hm", "PK", "IVP", "DMS", "IVD",
"NENISOIEC", "OG", "UVO", "PMS", "EDP", "MSS", "CM", "ULMA", "TI", "Q", "TIS",
"FEM", "DTM", "K", "PKKW", "VRI", "OWB", "N", "CSM", "S"]

pattern3 = r'\b(?:' + '|'.join(abbriviations) + r')\b'

def remove_abbriviations(text):
    return re.sub(pattern3, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_abbriviations)

#removal of residual words
other_words = "Prio", "halfVLS", "lip", "half", "Kelly", "Sloop", "itsos"]

pattern4 = r'\b(?:' + '|'.join(other_words) + r')\b'

def remove_other_words(text):
    return re.sub(pattern4, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_other_words)
```

Listing 16 Adjusted code for A15

```
#removal of residual abbreviations
abbriviations = "RKP", "KW", "AD", "hm", "PK", "IVP", "DMS", "IVD",
"NENISOIEC", "OG", "UVO", "PMS", "EDP", "MSS", "CM", "ULMA", "TI", "Q", "TIS",
"FEM", "DTM", "K", "PKKW", "VRI", "OWB", "N", "CSM", "S"]

pattern3 = r'\b(?:' + '|'.join(abbriviations) + r')\b'

def remove_abbriviations(text):
    return re.sub(pattern3, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_abbriviations)

#removal of residual words
other_words = "Prio", "halfVLS", "lip", "half", "Kelly", "Sloop", "itsos"]

pattern4 = r'\b(?:' + '|'.join(other_words) + r')\b'

def remove_other_words(text):
    return re.sub(pattern4, '', text, flags=re.IGNORECASE)

df['Name'] = df['Name'].astype(str).apply(remove_other_words)
```

Listing 17 Adjusted code for Twente Canals