KADIR HAS UNIVERSITY

SCHOOL OF GRADUATE STUDIES

PROGRAM OF MANAGEMENT INFORMATION SYSTEMS

A MACHINE LEARNING APPROACH TO STEEL SHEET PRODUCTION SURFACE QUALITY

ASENA ÖZTÜRK

MASTER OF SCIENCE THESIS

ISTANBUL, 2024

Asena Öztürk Master of Science Thesis 2024

A MACHINE LEARNING APPROACH TO STEEL SHEET PRODUCTION SURFACE QUALITY

ASENA ÖZTÜRK
ADVISOR: Assoc. Prof. Mehmet Nafiz AYDIN

A thesis submitted to
the School of Graduate Studies of Kadir Has University
in partial fulfilment of the requirements for the degree of
Master of Science in Management Information Systems

Istanbul, 2024

APPROVAL

This thesis/project titled A MACHINE LEARNING APPROACH TO STEEL SHEET PRODUCTION SURFACE QUALITY submitted by ASENA ÖZTÜRK, in partial fulfillment of the requirements for the degree of Master of Science in Management Information Systems is approved by

Assoc. Prof. Mehmet Nafiz AYDIN (Advisor)
Kadir Has University

Assist. Prof Tuğçe BALLI
Kadir Has University

Assist. Prof. Emine Elif TÜLAY
Muğla Sıtkı Koçman University

I confirm that the signatures above belong to the aforementioned faculty members.

Prof. Dr., Mehmet Timur AYDEMİR
Director of the School of Graduate Studies
Date of Approval: 31.12.2024

DECLARATION ON RESEARCH ETHICS AND PUBLISHING METHODS

I, ASENA ÖZTÜRK; hereby declare

- this Master of Science Thesis (or Graduation Project) that I have submitted is entirely my own work and I have cited and referenced all material and results that are not my own in accordance with the rules;
- this Master of Science Thesis (or Graduation Project) does not contain any material from any research submitted or accepted to obtain a degree or diploma at another educational institution;
- and that I commit and undertake to follow the "Kadir Has University Academic Codes of Conduct" prepared in accordance with the "Higher Education Council Codes of Conduct".

In addition, I acknowledge that any claim of irregularity that may arise in relation to this work will result in a disciplinary action in accordance with the university legislation.

Asena Öztürk

____

Date (31/12/2024)

To My Dearest Family, Coworkers and Academic Advisors...

ACKNOWLEDGEMENT

I appreciate the department of Management Information Systems in the faculty of Engineering and Natural Sciences at Kadir Has University for providing me this compelling studying opportunity. I am extremely grateful to my academic advisor Assoc. Prof. Mehmet Nafiz AYDIN for his encouraging support in my studies and research. I am also thankful to my fellow students in the MIS department and my colleagues for their support throughout this program. Finally, the biggest thanks that I want to give is to my family. They were always so supporting and loving.

A MACHINE LEARNING APPROACH TO STEEL SHEET PRODUCTION SURFACE QUALITY

ABSTRACT

The steel sheet production industry is of paramount importance, serving as the backbone for various sectors including construction and automotive manufacturing. A prevalent issue within this industry is the assurance of surface quality in steel sheets. Surface defects can lead to significant financial losses due to material rejections, increased processing costs, and potential failures in their end-use applications. Traditional methods for defect evaluation in steel sheet production, which rely heavily on manual inspection, are not only time-consuming but also limited in their ability to handle large data volumes and complex defect patterns. The integration of machine learning techniques presents a transformative potential to overcome these limitations. This thesis aims to develop a machine learning approach for defect evaluation in steel sheet production. The overall goal of this research is to improve the defect decision process by integrating human knowledge with technical (product related) data. The research adopts a case study approach by employing the data accumulated over 4 years. We conducted a literature review on steel surface defects, decision support systems, classification algorithms, and text mining. The study focuses on the detection and repair of defects, aiming to eliminate defects in production and optimize decisions related to defect detection and repair. The methodology of the study involves comparing different classification techniques and enhancing these results with text processing applications. The results for building a classification algorithm for the defect decision without including the textual data are not promising by giving around 30% error rate due to the multiclass nature of the problem. The study also concludes that the existence of text data improves the performance of the classification algorithms by decreasing the error rate to around 24%. These results show the significance of the textual data of the steel sheet defect decision process.

Keywords: Machine Learning, Defect Evaluation, Steel Sheet Production, Text Mining, Data-driven, Knowledge-based

RULO SAC ÜRETİMİ YÜZEY KALİTESİ MAKİNE ÖĞRENMESİ YAKLAŞIMI

ÖZET

Çelik sac üretim endüstrisi, inşaat ve otomotiv imalatı gibi çeşitli sektörlerin bel kemiği olarak büyük önem taşımaktadır. Bu endüstride yaygın bir sorun, çelik saclarda yüzey kalitesinin sağlanmasıdır. Yüzey kusurları, malzeme reddi, artan işleme maliyetleri ve nihai kullanım uygulamalarında potansiyel arızalar nedeniyle önemli mali kayıplara yol açabilir. Çelik sac üretiminde geleneksel kusur değerlendirme yöntemleri, büyük veri hacimlerini ve karmaşık kusur desenlerini ele almakta sınırlı kalmakla birlikte zaman alıcıdır. Makine öğrenimi tekniklerinin entegrasyonu, bu sınırlamaları aşmak için dönüşümsel bir potansiyel sunmaktadır. Bu tez, çelik sac üretiminde kusur değerlendirmesi için bir makine öğrenimi yaklaşımı geliştirmeyi amaçlamaktadır. Bu araştırmanın genel hedefi, insan bilgisini teknik (ürünle ilgili) verilerle bütünleştirerek kusur karar sürecini iyileştirmektir. Araştırma, 4 yıl boyunca birikmiş verileri kullanarak bir vaka çalışması yaklaşımını benimsemektedir. Çelik yüzey kusurları, karar destek sistemleri, sınıflandırma algoritmaları ve metin madenciliği üzerine bir literatür taraması yaptık. Çalışma, üretimde kusurları tespit etmeyi ve onarımını hedefleyerek, kusursuzluğu sağlamayı ve kusur tespiti ve onarımı ile ilgili kararları optimize etmeyi amaçlamaktadır. Çalışmanın metodolojisi, farklı sınıflandırma tekniklerini karşılaştırmayı ve bu sonuçları metin işleme uygulamaları ile geliştirmeyi içermektedir. Metin verisi dahil edilmeden yapılan kusur kararına yönelik sınıflandırma algoritması oluşturma sonuçları, çok sınıflı doğası nedeniyle yaklaşık %30 hata oranı vererek umut verici değildir. Çalışma ayrıca metin verisinin varlığının sınıflandırma algoritmalarının performansını artırarak hata oranını yaklaşık %24'e düşürdüğü sonucuna varmaktadır. Bu sonuçlar, çelik sac kusur karar sürecinde metin verilerinin önemini göstermektedir.

Anahtar Sözcükler: Makine Öğrenmesi, Kusur Değerlendirme, Çelik Sac Üretimi, Metin Madenciliği, Veri Odaklı, Bilgi Temelli Karar Alma

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

(Note: Table 1.1 refers to the first table in Chapter/Section 1, Table 10.1 refers to the first table in Chapter/Section 10 and Table A.1 refers to the first table in Annex A.)

LIST OF ACRONMYMS AND ABBREVIATIONS

AI: Artificial Intelligence

CNN: Convolutional Neural Networks

DSS: Decision Support System

KBDSS: Knowledge Based Decision Support System

KBIS: Knowledge Based Information System

KG: Knowledge Graph

LCI: Line Confocal Imaging

RNN: Recurrent Neural Networks

SVM: Support Vector Machines

XAI: Explainable Artificial Intelligence

ZDM: Zero Defect Manufacturing

1. INTRODUCTION

Given that companies try to achieve operational excellence in manufacturing, they aim to achieve a zero-defect production system in order to reach maximum capacity, minimum production cost and maximum profitability and also maintain customer requirements and competitiveness in the market. The concept goes back to in 1965 in the US army, the time of cold war (US Assistant Secretary of Defense 1965). After that it has become a promising approach for manufacturing industry. By the help of the emerging technologies supporting the Industry 4.0, the applicability of this concept in real life has become easier (Psarommatis et al. 2019, 1-15). However, production defects are inevitable problems of the process. Manufacturers implement different systems and processes to manage these defects and their consequences. In the process of preventing defects, technical information such as monitoring the production line, material qualities and production process variables, details of which can be found in the ISO 9000 standard (International Organization for Standardization 2015), is used. While all these are being followed, seven quality control tools (Magar et al. 2014, 364-371) are activated at different parts of the process for different purposes. While the aim is to reduce the number of defects that will occur with defect-preventing systems and processes during production, the decisions to be made after the defect are also of critical importance to prevent the increase in production costs.

Business, customer, and production process knowledge also comes to the fore in the decisions made after the defect. This shows that the knowledge of the person working in the production process is also included in the relevant decisions. Not only the data from the defective product, but also the situation of the order, the capabilities of the customer, or the afterwards processes on the product can cause a different decision on the same defect condition. This knowledge comes from the history of defect data and the notes or reports related to these defects. This data is mostly unstructured or semi-structured like text or pictures. With the help of the digitalization this kind of experience-based knowledge is more accessible by computer analysis in recent years. Therefore, manufacturers want that information to be useful in defect decision processes. The increase in social media usage has already pushed companies to use this data and, since then various application areas has emerged like, digital libraries, resume filtering or business intelligence areas (Preeti 2021, 474). Nowadays manufacturers have already made some digital and technologic investments on the process control part of the production with the smart factory initiative (Chen 2022, 2). Therefore, for the next step in the data-driven decision-making process is embedding the knowledge to the digital data. One of the most common unstructured knowledge data is texts from reports or notes and these can be used for defect analysis (Wang et al. 2024, 17).

Given the complexity and variability inherent in manufacturing processes, traditional methods of defect evaluation can be insufficient and time-consuming. The integration of a machine learning approach can significantly enhance the accuracy and efficiency of defect detection and decision-making by leveraging both structured technical data and unstructured human knowledge from text reports. This approach enables a more holistic analysis, providing deeper insights and predictive capabilities that can lead to more informed and timely responses to defects, ultimately reducing costs and improving production quality.

This thesis aims to develop a machine learning approach for defect evaluation process by focusing on the text data analysis. The main concern of this approach will be improving the defect decision process by integrating the human knowledge to the technical data of the defective product. The machine learning approach will be developed and evaluated with a case study that involves quality defect decision data from 2020.

1.1 Background and Context of the Partner Company

Borçelik, established in 1990 as Turkey's pioneering private and its second-largest flat steel producer, commenced operations to create "cold-rolled steel rolls" in 1994. With investments totaling 530 million USD in 1994, 2003, and 2008, Borçelik expanded its production capacity to 1.5 million tons. Boasting the largest galvanized steel production in Turkey, Borçelik operates three cold-rolling and three hot dip galvanized steel lines, delivering top-tier quality across a galvanized steel capacity of 900 thousand tons.

The robust market standing of the firm, a joint venture between Borusan Holding and ArcelorMittal, one of the world's leading steel producers, is underpinned by its dynamic workforce, innovative spirit, ongoing investments for self-improvement and expansion, and a commitment to customer-focused service and quality.

Operating from its 240,000 square meter facility in Gemlik, Borçelik conducts manufacturing in hot dip galvanized steel, cold-rolled steel, and hot-rolled (pickled and oiled) steel categories, all essential industrial raw materials. Borçelik's production includes a range of steel types such as commercial, drawing, deep drawing, extra deep drawing, bake-hardening, dual phase, rephosphorized, HSLA (high strength low alloy), high-carbon, enameling, and structural steel. The annual production capacity of Borçelik totals 1.5 million tons, consisting of 600 thousand tons of cold-rolled and 900 thousand tons of hot dip galvanized steel.

Borçelik is committed to sustainable production in its hot dip galvanized steel, cold rolled steel, and hot rolled steel (pickled and oiled) categories, all vital as industrial raw material inputs. Hot dip galvanized steels are flat steels that undergo a continuous hot dip process for zinc/galvanized coating to prevent corrosion. The steel is chemically bonded to the zinc/galvanized coating after being immersed in a molten zinc pot, providing resistance to corrosion and cathodic protection. Cold-rolled steels are produced by cold rolling pickled and oiled hot-rolled flat steels, with "Cold-Rolled Steel" resulting from surface cleaning, recrystallization annealing, surface roughening, and tempering post-rolling. Borçelik caters to specific end-user needs with its diverse qualities and grades derived from cold-rolling, annealing, and tempering. The other product, hot-rolled (pickled and oiled), involves removing the oxide scale layer from the steel surface with hydrochloric acid (pickling) and applying oil to prevent corrosion (oiling), resulting in "Pickled and Oiled Hot-Rolled Steel".

Borçelik integrates into every aspect of life, providing services to sectors such as household appliances, automotive (both main and subsidiary industries), panel radiators, construction, color coating, pipes & profiles, packaging, metal goods, and steel service centers. (Borcelik, 2024)

1.2 Objectives and Structure of Thesis

Steel is one of the most significant materials in global manufacturing due to its reusable and repairable properties. However, the manufacturing processes involved are both costly and intricate. Additionally, steel boasts a lengthy history, and knowledge regarding its production and application is highly specialized, both in the market and within the partner company. Given the substantial costs associated with defects in steel, companies take defect management very seriously. Options for addressing defects include repair or delivery to the customer despite the defect, and decisions in this context require substantial technical and business expertise. Beyond product technical data, considerations such as customer plans, capabilities, and intended use of the product are crucial in the defect evaluation process (Wen et al. 2023, 4).

The integration of a machine learning approach in manufacturing processes can significantly improve defect detection and decision-making by combining structured technical data with unstructured human knowledge from text reports. This holistic analysis offers deeper insights and predictive capabilities, leading to timely and informed responses to defects, ultimately reducing costs and enhancing production quality.

This study will evaluate defect data and the corresponding decisions to develop a machine learning approach for defect evaluation. Initially, the analysis will exclude textual data to create a prediction model for defect assessment. This segment of the overarching problem will be treated as a classification problem, and various algorithms will be implemented and compared. During the development of these models, a feature selection step will be incorporated. The analysis will be further refined by incorporating text data columns as binary variables. The impact of this integration will be assessed and compared with the initial analysis. Subsequently, the text data will be examined in their original form. Through text mining applications, these columns will be analyzed both individually and collectively as a single column. All analyses will be conducted using KNIME software. These steps will explore the influence of decision notes on the evaluation process.

The primary objective of this research is to analyze and compare both tacit knowledge data and structured data derived from steel surface defect decisions by employing appropriate algorithms to predict a unified target variable. Through these analyses, the study aims to propose a machine learning approach along with potential enhancements. This approach addresses a specific requirement articulated by the partner company; hence, the research will serve as a foundation for further studies and application projects. This study will contribute to the problem-solving process for the partner company, acting as an initial step in this endeavor.

There are two major steps in this study that investigates the below questions:

- Can a defect decision be made by looking at the historical defect data and decision on the defect?
- Can the textual data, gathered from defect notes by relative workers be used to make a defect decision?
- How can these data be combined in one machine learning model?

These questions guide this study towards understanding the partner company’s concerns from a machine learning application perspective.

Initially, the literature on similar application studies will be reviewed. These studies will be evaluated in terms of problem definition, application area, methodology, and key findings, and will be compared to one another. This study will also be incorporated into this comparison, with the major differences explained.

The following sections of the thesis will describe the data, the preparation process, the classification problem, and the text mining approach to the problem. After completing these components, the research will present clear conclusions and propose directions for future work.

2. LITERATURE REVIEW

Developing technologies effect every aspect of our lives. So, it is impossible to be away from these changes in manufacturing industries too. Production quality or maintenance efficiency is at the scope of the digital transformation that is started with the aim of developing and increasing sales at the beginning. Industrial revolution has brought new kind of data with the help of the new technological devices. As a result of that data driven decision opportunities increased in those areas. Optimizing production process, reducing cost, and improving efficiency are some examples of these opportunities (Wang et al. 2020, 2).

Steel remains a vital material in various industries, with applications ranging from civil engineering to household tools. The World Steel Association shares the global data about this industry as being quite stable from 2020 at around 1900 million tones total production per year (World Steel Association 2024). As steel quality directly impacts product and infrastructure quality, controlling steel production quality remains essential for ensuring qualified products (Wen et al. 2023, 1). Therefore, the quality of steel products becomes a major application area of the technological developments. For both commercial or academic experts try to develop solutions or applications for the needs of quality assurance at steel manufacturing industry.

This study inquires the literature of technological or process applications on manufacturing, especially steel manufacturing, in order to have a common sense of the general problems in the scope. The quality part of the manufacturing is mostly in concern of this study. Some papers propose a software for defect problem, others focus on the defect evaluation process and decision making points in this process. While doing this research it is seen that, most problem definitions refer to the prevention of the defects at the production process. This leads to image processing technologies or machine learning algorithms to detect the defects faster and take actions about this situation (Wen et al. 2023, 4-23). Since, another aspect of defect detection is the cost of the repair, it is an expected case to focus on prevention of defects. Lots of studies can be found in literature about these subjects. Selected seven examples shows the variety of the application areas, methodologies or achieved results can be summarized like following:

- Fault Location of Strip Steel Surface Quality Defects on hot-Rolling Production Line Based on Information Fusion of Historical Cases and Process Data (Wang et al. 2020, 3) defines the main problem as the challenge of accurately locating faults on the hot-rolling production line due to surface defects of strip steel. The application area of this study is Steel manufacturing, particularly in the process of hot-rolling strip steel. The study presents a model that combines historical data and process information with categorizing defect causes and fuzzy semantic reasoning, assessing feature significance using XGBoost, to efficiently trace and locate faults in hot-rolling lines, enhancing expert decision-making and fault source identification.

- Evidence-Based and Explainable Smart Decision Support for Quality Improvement in Stainless Steel Manufacturing (Tiensuu et al. 2021, 16-19) addresses the need for effective process control to prevent defective products and reduce failure costs, emphasizing the significance of data mining and machine learning methods in quality control and decision-making processes in stainless-steel strip manufacturing. The paper shows how production line parameters affect real-time roughness and how machine learning with data collection using LCI technology for no-contact measurement, surface roughness prediction using generalized boosted regression, and XAI can help with data quality and problem solving in manufacturing.

- Data Mining for Fault Diagnosis in Steel-making Process under Industry 4.0 (Chen 2022, 7-8) tackles the problem of detecting and forecasting failures in steel production using data-driven methods to handle the complex and high-dimensional processes. The paper presents a data-driven framework for defect diagnosis and prediction in steel-making, using semantic technologies and Knowledge Graphs to fuse multi-sourced data and knowledge.

- A Hybrid Decision Support System for automating decision making in the event of defects in the era of Zero-Defect Manufacturing (Psarommatis and Kiritsis 2022, 2-3) aims to address this issue by implementing a hybrid Decision Support System (DSS) that combines data-driven and knowledge-based to automate decision-making processes for defect repair at the manufacturing industry, specifically targeting the Zero-Defect Manufacturing (ZDM) strategies. This study presents a hybrid DSS that combines data and knowledge using the MASON ontology to enrich data with context and analyzing defects using real-time and past data. It also evaluates and suggests repair plans based on dynamic criteria.

- A decision-making model for the rework of defective products (Soares et al. 2019, 69) revolves the problem definition around the need for a decision-making model that can help determine whether to discard or rework defective products, with the goal of reducing the cost of defects and overall quality costs in mass production industries. The study introduces a decision-making framework that weighs different costs for flawed items, with options to discard or rework, with or without substitution. It utilizes decision flow and tree analysis, and confirms the model's accuracy through data gathering, evaluation, comparison, sensitivity analysis, and discussion.

- Knowledge Based Decision Support System in Steel Industries (Sharma and Kumar 2021, 1-3) discusses the challenges faced by the steel industry in decision-making due to the complexity of operations and the need for expertise in multiple fields within steel plants. The problem defined is the difficulty in making timely and accurate decisions due to the extensive knowledge required, which is often gained through long experience. The study emphasizes the importance of Knowledge-based Information Systems (KBIS) and Decision Support Systems (DSS) in aiding decision-making and gaining knowledge of best business practices within the steel industry. It discusses the challenges and opportunities of KBDSS in manufacturing industries.

- DuAK: Reinforcement Learning-Based Knowledge Graph Reasoning for Steel Surface Defect Detection (Zhang et al. 2023, 2) defines the problem of steel surface defect detection as a crucial factor affecting the product quality of steel products. Current research primarily concentrates on identifying and categorizing defects through machine vision-based algorithms, which overlooks the tracing of potential causes and the reuse of experiential knowledge. The study involves constructing a knowledge graph with a policy-driven reinforcement learning algorithm by industrial data. To navigate the graph efficiently, two oppositely directed agents are deployed. An integrated reward function is utilized, considering the direction and length of the path, as well as the distance between entities, to guide action selection. Furthermore, a path sharing mechanism, and an updated selection policy are adopted to capitalize on previously acquired knowledge.

The focus of this study, which is the after-defect evaluation process, serves the other part of the literature. Also, with this study knowledge acquisition implementation is tried which focuses on software part of the decision making, not the process part. An improved classification of defects with text data provided by the production or quality operators is proposed by this study. This machine learning model is an example of hybrid models in terms of the data source types that is used. Moreover, the proposed solution for the related problem is an example of text processing usage in manufacturing environments instead of sales or customer experience area. The future scope of this study includes enhancing the text data and joining visual data to increase the hybrid feature of model and gather more accurate results, due to the request for combining digital data with human information. For the time being, this study examines the effect of the text data on a classification problem.

2.1 Background and Context of Surface Quality in Steel

Steel stands as a highly prevalent material across numerous sectors, including construction, automotive, aerospace, and manufacturing. However, steel products often suffer from surface defects that can degrade their quality and performance. Surface defects are irregularities or discontinuities on the steel surface that result from the manufacturing process, such as rolling, casting, forging, or welding. Some common types of surface defects are cracks, scratches, pits, scales, inclusions, seams, and roll marks (Hao et. al. 2020, 1834-1835).

Surface defects can have negative impacts on the mechanical properties, corrosion resistance, fatigue life, and aesthetic appearance of steel products that can be crucial due to the usage area of the product. Therefore, it is essential to detect and classify surface defects accurately and efficiently, and to take appropriate corrective actions to prevent or reduce them. Surface defect detection and classification can be done by various methods, such as visual inspection, ultrasonic testing, eddy current testing, magnetic particle testing, and thermography (Wen et al. 2023, 4-24). However, these methods have some limitations, such as high cost, low speed, low sensitivity, and human error.

Recently, machine learning techniques have been applied to steel surface defect detection and classification, using image processing and computer vision algorithms. Machine learning techniques can automate the defect detection and classification process, and improve the accuracy, speed, and reliability of the results. Machine learning techniques can be divided into two categories: supervised and unsupervised. Supervised techniques require labeled data for training and testing, while unsupervised techniques do not (Wang et al. 2020, 11).

Machine learning techniques for steel surface defect detection and classification have shown promising results in terms of accuracy, robustness, and efficiency. However, there are still some challenges and limitations, such as the availability and quality of data, the choice and optimization of parameters, the generalization and scalability of models, and the interpretation and explanation of results. Future research directions may include the integration of multiple machine learning techniques, the use of transfer learning and domain adaptation, the incorporation of prior knowledge and expert feedback, and the development of explainable and interpretable machine learning models (Wang et al. 2020, 11).

2.2 Background and Context of Classification Algorithms

Classification algorithms are a cornerstone of machine learning and data science, playing a pivotal role in numerous applications ranging from medical diagnosis to spam detection. This literature review explores the fundamentals, types, applications, and advancements in classification algorithms, supported by relevant scholarly sources. Classification algorithms are supervised learning techniques that categorize data into predefined classes. These algorithms learn from labeled training data and make predictions for new, unseen instances. The primary goal is to develop a model that can accurately assign labels to new data points. According to James et al. (2013), classification is a fundamental task in statistical learning, involving the construction of decision boundaries that separate different classes in the feature space. The typical evaluation of classification algorithm performance involves metrics like accuracy, precision, recall, and the F1-score, which measure the algorithms' effectiveness in correctly classifying data. Classification algorithms can be broadly categorized into several types, each with its unique characteristics and applications.

Linear Classifiers: Linear classifiers, such as logistic regression and linear discriminant analysis, assume a linear relationship between the features and the target variable. They are simple and computationally efficient, making them suitable for linearly separable data (Hastie et al. 2009, 79-103).

Tree-Based Methods: Decision trees, random forests, and gradient boosting machines are popular tree-based methods. These algorithms split the data into subsets based on feature values, forming a tree-like structure. Tree-based methods are versatile and can handle non-linear relationships and interactions between features (Breiman 2001, 5-6).

Support Vector Machines (SVM): Support Vector Machines (SVMs) are designed to identify the best possible hyperplane that separates different classes with the widest margin. They perform well in spaces with many dimensions and are capable of both linear and non-linear classification, thanks to kernel functions that transform the data. (Cortes and Vapnik 1995, 273-277).

Neural Networks: Neural networks, especially deep learning models, have become prominent for their capacity to discern intricate patterns and connections within data. Convolutional neural networks (CNNs) are commonly utilized for image classification, while recurrent neural networks (RNNs) are preferred for sequence classification tasks. (LeCun, Bengio, and Hinton 2015, 436-442).

Bayesian Methods: Bayesian classifiers, such as Naive Bayes, utilize Bayes' theorem to compute the probability of each class given the features. These methods are particularly useful for text classification and problems involving probabilistic reasoning (Mitchell 1997, 154-199).

Classification algorithms are applied across various domains, demonstrating their versatility and effectiveness from medical diagnosis to spam or fraud detection. Recent advancements in classification algorithms focus on improving accuracy, scalability, and interpretability.

Classification algorithms are essential tools in the machine learning toolkit, offering solutions to a wide range of problems. From linear classifiers to deep learning models, these algorithms continue to evolve, driven by advancements in technology and the growing demand for accurate, scalable, and interpretable models. The ongoing research and development in this field promise even greater capabilities and applications in the future.

2.3 Background and Context of Text Mining

Text mining is the process of extracting useful and meaningful information from unstructured text data. Text mining has various applications in different domains, such as natural language processing, information retrieval, sentiment analysis, topic modeling, and text summarization (Feldman and Sanger 2007, 276-309). Text mining can also be used to analyze and understand the content and quality of academic literature, which is essential for researchers and scholars who need to keep up with the latest developments and trends in their fields.

Text mining is the process of uncovering significant patterns and insights from textual data sources. It intersects various disciplines such as information retrieval, data mining, machine learning, statistics, and computational linguistics. Employing techniques like summarization, classification, and clustering, text mining extracts valuable knowledge from natural language texts in semi-structured and unstructured formats. It has applications across diverse fields including industry, academia, web services, and social media, aiding in tasks like opinion mining, feature extraction, sentiment analysis, predictive analytics, and trend analysis for systems like search engines, customer relationship management, email filtering, product recommendation analysis, fraud detection, and social media analytics (Talib et al. 2016, 416-417).

Text mining has various steps that can be grouped into four main parts (Preeti 2021, 474).

- Data collection: This step involves gathering text data from various sources and formats, such as plain text files, web pages, pdf documents, and others.

- Data cleaning: This step involves removing stop words (words that do not carry much meaning), applying stemming (the process of finding the root of a word), and indexing the data to capture the essence of the text.

- Data processing: This step involves applying automatic methods to check and improve the quality of the data using techniques such as normalization and standardization.

- Data extraction: This step involves using the processed data to find relevant and meaningful patterns and insights for decision making and market analysis.

Text mining techniques are broadly classified into nine groups: identifying entities, navigating through text, searching and retrieving, clustering, categorizing, summarizing, analyzing trends, associating, and visualizing. Entity extraction pinpoints specific details within text, such as recognizing noun phrases that denote individuals, locations, or organizations. This includes extracting terms and measuring their frequency (keyword frequency). Text-based navigation helps in contextualizing related terms and mapping their relationships. Search and retrieval allow for locating pertinent information using set search parameters. Clustering involves grouping similar documents to maximize intra-group similarities and minimize inter-group similarities. Categorization organizes raw data into predefined topics for analysis using content-mining technologies. Summarization condenses document content, reducing the reading required. Trend analysis identifies patterns in time-sensitive textual data. Association analysis connects one pattern with another. Visualization uses feature extraction and key term indexing to create graphical representations, highlighting main topics or concepts and facilitating document location within a graphical representation. Text mining's complexity ranges from simple (e.g., arithmetic averages) to intermediate (e.g., linear regression, clustering, decision trees), to highly complex methods like neural networks (Mohammad and Alkin 2007, 5).

3. MACHINE LEARNING APPROACH FOR DEFECT EVALUATION

This section of study will evaluate the data provided from the partner company Borçelik and apply possible classification models on this data. This data consists of the product and defect information for the decisions made on this defect. Textual notes on defect or decision are found in data set. Since the partner company has a commercial business, details about data that in use will be given limitedly. Finally, a comparison of the results from these models will be interpreted.

3.1 Data Analysis

As being the owner of the main problem of this study, Borçelik Metallurgical Quality Department shares its defect evaluation data in order to be studied. For data analysis KNIME is used since it is easy to use, and results of the filters or modifications can be quickly obtained. This data contains detailed product, customer, defect, and decision from the year 2020 as a table with 393.393 rows and 85 columns. 23 of these columns contains numeric data while the other 62 columns have string type of data as can be seen in Table 3.1.1.

Table 3.1.1 Data type summary of starting data set

Data Type	**Count**
Numeric	23
String	62
Total	**85**

Those 85 columns give different information about the defective product. These information can be grouped into 7 sections. Production section gives the information about production steps or parameters for that product. Physical, chemical or mechanical properties of the product is another group of information. In addition to those, raw material information is also given in the data set. This group consists quality, vendor, or purchasing information of the raw material. Even the manufacturer id of the product, purchasing order or the ship that is used in transportation is an information for raw material. The key point of the data is grouped in defect section. In that group, defect code, number, explanation, dates and the user of the defect record, root cause explanation, or the notes about defect can be found. Target customer, distribution channel, or material information is grouped as customer related information. The main investigation point of the data is defect decision section of these groups. In that section decision code, explanation dates and the user of the defect decision record, and the notes on defect decision is grouped. Last group has two columns that shows if there is a claim data for that defect. The groups of data columns can be seen below at Table 3.1.2. Target column is defect decision code that is in the group that is called defect decision. Like the starting point of this data set, defect code, target column has no missing values. There are 15 different values in that column that is the main concern of the classification problem of this study. These values have different meanings in production or shipment processes. Decision on defect can be continue as it is or change its order for another customer due to the suitability of the product. Repair or adding extra process on product can be another decision for the defective product. For all those defect decision code there is an explanation, given by the quality operator, consists of information about production, quality or customer properties of the target product.

Table 3.1.2 Data type summary of starting data set

Group	**# of Columns**
Production	26
Product	17
Raw Material	17
Defect	9
Customer	8
Defect Decision	6
Claim	2

Some descriptive statistics values can be calculated to deep dive into the numeric data. Min, max, mean, and similar other statistics with the number of missing values are starting information for data analysis. In this data some missing values have a meaning for the process therefore, all missing values cannot be evaluated same. That distinction of missing value handling strategies are determined by consultancy from the accountable department. Although the columns of the data defined as string or numeric, the contents of the columns have other meanings for the business. For example, customer numbers or product numbers are defined as numeric data, however, they should be considered as a categorical data. On the other hand, some numeric data such as weight of the product are irrelevant to the defect decision. Therefore, an elimination process had been done on data with problem owner regarding to their business knowledge. The detailed statistics and missing value information about numeric data is shown in Table 3.1.3. NUM3, NUM8 and NUM9 columns are removed due to the uniqueness for each row or relevancy issues. NUM6 and NUM7 columns are removed because of the excess number of missing values. On the other hand, the other missing values are meaningful for the process and the columns are kept. Another aspect of numeric data is that values representing a category. This situation is valid for NUM7 and NUM20 columns. For column NUM7 last four digits stands for a category and data is manipulated by that rule. Whereas NUM20 is a column that the missing values have a meaning. Therefore, another data manipulation rule is applied on this column that categorizes the data as blank ones and the others.

Table 3.1.3 Descriptive Statistics for Numeric Data

Column	**Min**	**Max**	**Mean**	**Std. deviation**	**Variance**	**Skewness**	**Kurtosis**	**No. missings**
NUM1	-	625	0	1	2	444	196.694	-
NUM2	-	25.862	1.612	1.563	2.441.792	2	5	-
NUM3	250	32.330	13.714	6.781	45.982.543	0	-1	-
NUM4	4	18	13	3	11	-0	-1	1.455
NUM5	1.000.002	2.228.113	1.259.655	430.871	185.650.173.438	1	-0	1.455
NUM6	4	24	16	5	21	0	-1	389.436
NUM7	100.228	9.839.700	1.134.099	434.157	188.492.518.787	11	176	390.150
NUM8	180.001.657	180.008.434	180.006.537	1.895	3.589.845	-1	0	393.353
NUM9	1.599.143	2.185.046	1.896.144	170.202	28.968.856.707	-0	-1	-
NUM10	1	96	58	29	854	-1	-1	257.766
NUM11	665	1.565	1.185	183	33.499	0	-1	300.838

NUM12	1	2	2	0	0	-13	168	393.051
NUM13	2	2	2	-	-	-	-	393.241
NUM14	1	3	2	0	0	1	27	392.333
NUM15	1	3	2	0	0	-2	194	390.040
NUM16	1	8	2	0	0	3	86	391.580
NUM17	1	2	2	0	0	-34	1.142	392.251
NUM18	1	5	2	0	0	11	213	389.780
NUM19	1	5	2	1	1	0	-1	258.175
NUM20	420.322.012	2.970.943.001	2.114.971.313	591.536.521	349.915.456.638.116.000	-1	-0	347.851
NUM21	1	3	2	1	0	0	1	390.209
NUM22	-	500	0	5	26	58	4.083	-
NUM23	-	385	0	4	17	54	3.355	-

String data in this data set is mostly about decision or defect definitions, notes, user that act, or text for some numeric attributes. Numeric attribute texts are removed at the beginning of the analysis because they are represented by another column. In addition to that, defect or decision time data is stored in string format. Since the main problem is not related to the time data, these columns are eliminated also. Moreover, user information for both defect and decision columns are removed. Again, unique row values are also eliminated from the dataset. Another column filtering strategy resulted from fully missing attributes or all same values due to the lack of effect on the result. Eight columns that has any value is excluded from the data set. Finally, there are some columns is data that can be seen as long texts and include mostly the notes on the process. This study aims to take these data into account at the next sections and they are removed from the starting data set. Information about all string columns is given in Table A.1 at the appendix section. The final number of columns in data set becomes 52 and details of data types are given in Table 3.1.4.

Table 3.1.4 Data type summary of filtered data set

Data Type	**Count**
Numeric	18
String	34
Total	**52**

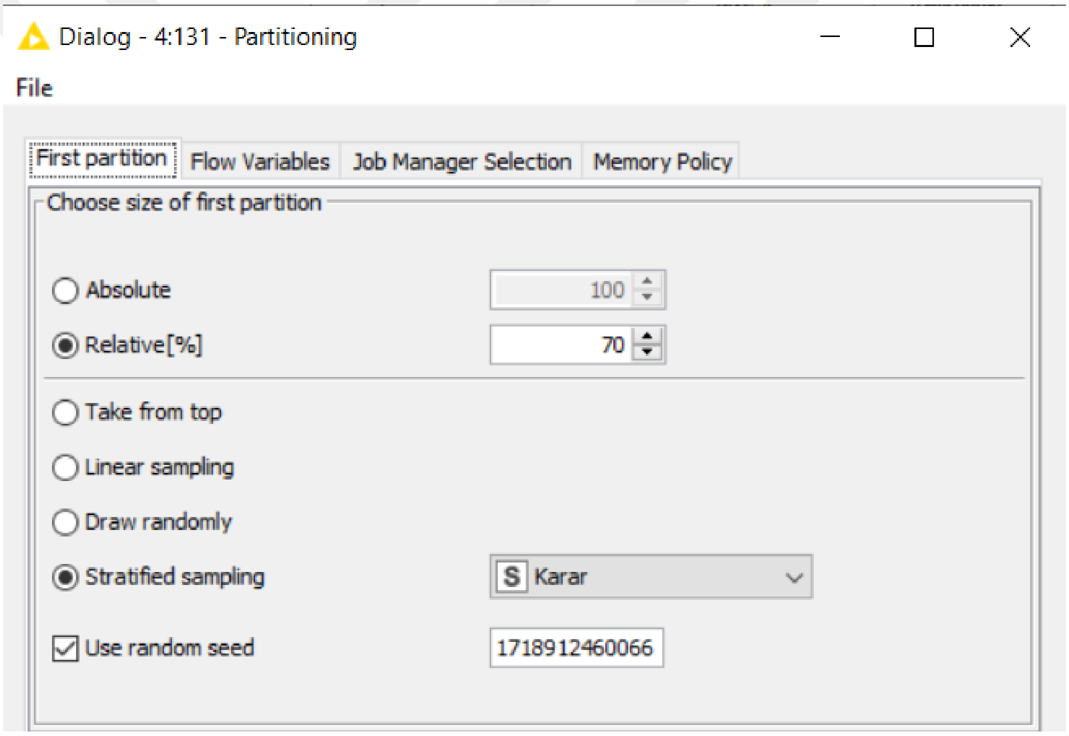
Exclusion of columns is the first step of data preparation. After that, all columns are processed for the missing values since the null data can cause inconsistencies at models that will be used on dataset. Missing value handling is done by replacing with ‘X’ in string columns and placing zeros in numeric columns.

Eventually all numeric data is changed to string type for being useful in classification problem. However, far more analysis is needed to achieve a cleaner data set. Unique values over 1000 take attention at the first sight. For two columns binning methodology is applied and number of unique values are reduced to less than or equal to 20. These columns have information about the dimensions of the product. In addition to that application, columns with both text and numeric data are taken into account. Since data has values from 2020, some attributes have changed over time and started to refer differently. Or the reporting mechanism has changed, and data started to come in text format rather than numeric values. With the help of the business knowledge, six columns manipulated to singular text equals of the numeric values. The KNIME workflow for data analysis and manipulation process is placed in appendix section as Figure A.1.

3.2 Classification Model Building

In this section of the study filtered and modified data set will be processed in four different models. Gradient Boosted Trees, Decision Trees, Random Forest, and Logistic Regression models, which are declared in the literature review part of this study as in the first two categories of classification algorithms, are parallelly placed in workflow. These algorithms are chosen because they balance predictive power, interpretability, and robustness compared to alternatives like SVM, Naive Bayes, or k-NN. Also these models are much more efficient in large data sets with their cheapness in computation. In summary, other models such as SVM, Naive Bayes, or k-NN might excel in certain niche applications, but respective chosen models provide a better balance between accuracy, flexibility, and computational efficiency for general classification tasks (Hastie et al. 2009, 9-40). All nodes in the workflow that has a parameter for stratified sampling is set with the same random seed value in order to make comparison of models in same circumstances and achieve same reprocess results of processes.

Parameters of the nodes used in KNIME standards in order to keep track of the nodes in big workflows. For example, when data partitioning is needed in workflow, it is made by stratified sampling of target column and with relative 70% value in standard setting as can be seen at Figure 3.2.1. Also, same random seed value, that is proposed by KNIME itself, is used in all nodes. All the node parameters, that are modified, are given in the appendix section of this paper after the respective model examples. Besides of all self-developed models, KNIME AutoML feature is also used to have another insight about data and possible outcomes. AutoML of KNIME is a component for supervised classification. It automatically runs a machine learning cycle with preparing data, optimizing parameters.


Dialog - 4:131 - Partitioning

Choose size of first partition

○ Absolute 100

● Relative[%] 70

○ Take from top

○ Linear sampling

○ Draw randomly

● Stratified sampling S Karar

☑ Use random seed 1718912460066

Figure 3.2.1 Partitioning parameter settings

The first section of workflow aims to find out the effect of the data columns on the target column. For this aim, feature selection loop ability of KNIME is used. All four feature selection models are processed in a loop for 50 times. KNIME feature selection loop selects columns randomly and runs models with these columns. The target column is included in all iterations. Only Gradient Boosted Trees model is processed with row sampling because of the performance issues related to the hardware. The other models included all data rows. Since the next section of this study will investigate the text related data, this section inquires the effect and performance of the classification of the non-text data columns and used the filtered 52 columns that mentioned before. Same workflows are run for both error and cohen’s kappa metrics. At the end of the loop, the score parameter of models is set to minimize the error or maximize cohen’s kappa.

After study on non-text data, in order to see a basic effect of text data, these related columns are added to the data set as binary data. If these columns have data in rows, they are changed to 1 and others have become 0. This modification is made to see the effect of the presence of any text data on classification problem. This effect is questioned by three aspects. The first aspect is to study the data that have any text data. In order to do that, all six text related columns, that are now binary, are filtered to be zeros. This filtering process decreases the number of rows to 6469. The aim of this part was to see if the models perform better at the absence of any text data. The second aspect is the reverse of the first one. When all six columns are filtered to be one, only 3 rows remained because of one feature. Therefore, that filtering is eliminated, and the data used in this aspect of analysis became a table with 18252 rows. The purpose of this analysis again to see the model performance change with data supported fully with text input. By making these two analyses, the whole data is examined in two pieces and finally with the third aspect the combined results are obtained. The third aspect is to use the data without filtering. It is still a basic analysis that gives a result about the effect of having a text data on models. Same workflows applied while doing this analysis. One workflow example is added at the appendices section of this study as Figure A2.

3.3 Classification Model Comparison

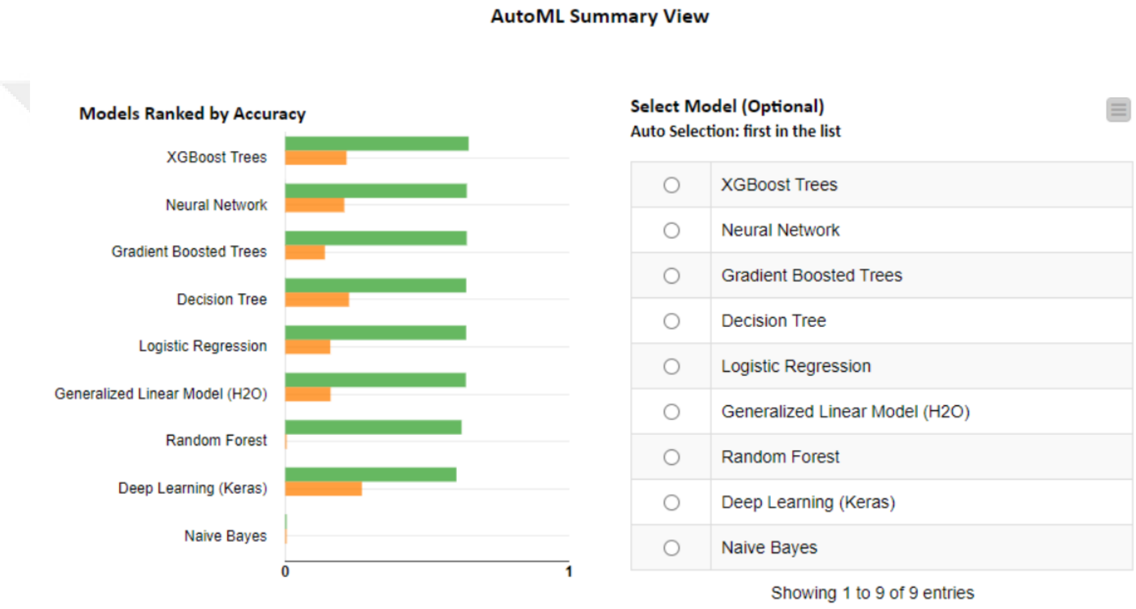
Data and effect of text data presence analysis resulted lots of classification model workflows. All models examined regarding two metrics: error and cohen’s kappa because feature selection process in KNIME processes data with respect to only three comparison metrics. These are accuracy, error and cohen’s kappa. Accuracy and error are metrics that measures the same parameters in inverse order. Therefore, using one of them is enough for analysis. On the other hand, cohen’s kappa is a different metric from those two and gives another aspect for measurement. Five classification workflows, including automl, compared with these metrics in four aspects: non-text data, all data with no text input, all data with text input and, all data including binary text data. In addition to specifically studied algorithms, automl applied other models to the data. Related values from the models are at Table 3.3.1.

Table 3.3.1 Score Metrics of KNIME Workflows

Analysis Aspect	Text Colums Excluded		All Text Columns Empty		All Text Columns Full		Data with Binary Text	
Model	Used Feature Count **Error**	Used Feature Count **Cohen’ s Kappa**	Used Feature Count **Error**	Used Feature Count **Cohen’ s Kappa**	Used Feature Count **Error**	Used Feature Count **Cohen’ s Kappa**	Used Feature Count **Error**	Used Feature Count **Cohen’ s Kappa**
Gradient Boosted Trees	45 **0.315**	50 **0.323**	56 **0.270**	56 **0.600**	56 **0.358**	56 **0.464**	54 **0.256**	52 **0.517**
Decision Tree	30 **0.334**	36 **0.359**	48 **0.300**	48 **0.554**	48 **0.417**	48 **0.405**	54 **0.251**	54 **0.551**
Random Forest	50 **0.311**	44 **0.268**	51 **0.278**	51 **0.582**	49 **0.369**	49 **0.430**	52 **0.237**	51 **0.582**
Logistic Regression	50 **0.328**	50 **0.293**	50 **0.310**	50 **0.534**	52 **0.367**	52 **0.447**	57 **0.267**	57 **0.473**
AutoML	**0.354**	**0.223**	**0.297**	**0.560**	**0.420**	**0.353**	**0.266**	**0.485**

As can be seen from the results, when there is no text information in data all models give better results than the full data. On the other hand, all text columns have values data rows have better cohen’s kappa result opposite of the error rates. This part of data is much smaller due to the high number of missing values in those columns. Therefore, these results can be seen expected. When text columns are added to the equation, both measurement metrics rates of the models dramatically change. This shows that, even the existence of an extra information about target column has lots of influence on the prediction results.

AutoML node of KNIME compares ten different models and selects the best resulted one. In this case, one of the models is excluded by the component due to being not applicable on data set. Parameter optimization process is embedded for all models, but feature selection is not done in that component. All features are used, and the results are compared, and best model is chosen automatically. An example of the visualization given by the software for the results of the models can be seen at Figure 3.3.1.



AutoML Summary View

Select Model (Optional)
Auto Selection: first in the list

○	XGBoost Trees
○	Neural Network
○	Gradient Boosted Trees
○	Decision Tree
○	Logistic Regression
○	Generalized Linear Model (H2O)
○	Random Forest
○	Deep Learning (Keras)
○	Naive Bayes

Showing 1 to 9 of 9 entries

Figure 3.3.1 The model result visualization taken from KNIME AutoML

Since the other models are evaluated with two metrics, the same metrics are used for AutoML results. Results of the models are close to each other except for the last concept of modelling which is including text data in a binary mode to the categorical data. Best results belong to Random Forest algorithm in that particular concept of data usage. The error rate of the model drops one point while cohen’d cappa increases more than three points with inclusion of textual data existence information. AutoML brings the XGBoost algorithm for being the most accurate. One more analysis can be done by using XGBoost with feature selection process to see another example.

4. TEXTUAL DATA ANALYSIS

This section of study will evaluate the text related data by applying possible text mining models on this data. Finally, a comparison of the results from these models will be interpreted.

4.1 Data Analysis

Until this section, the data analyzed does not have any text related component. In this section text data is discussed and tried to see the effect on the decision of the defect. For this purpose, six text data columns are used in these analyses with different point of views. First of all, six text data columns are combined as a column and the model is designed on that. In addition to that, to see all individual effect of the data on the result, six parallel models are built, and the results are discussed.

Text data columns differ from each other with respect to their missing values or lengths. For example, one column is directly related to the target column and has no missing value while three columns are mostly missing. The detailed information about these columns can be seen at the Table 4.1.1.

Table 4.1.1 Text Columns Information

Text Columns	No. Missing	Unique Values	Avg. No. Char
STR1 (Defect Note)	65031	>1000	55
STR2 (Root Cause)	212521	366	8
STR3 (Decision Note)	213024	>1000	20
STR4 (Document)	225773	11	8
STR5 (MQ Note)	388937	787	0
STR6 (Decision Explanation)	0	>1000	25

This table show that the data that is wanted to be used is unbalanced, and it can affect the analysis result. STR2 values are mostly standardized that unique value number and missing value number are relatively less than the other columns. On the other hand, STR1, STR3 and STR6 are the operator notes on defect or defect decision and root cause explanation of the defect. Therefore, those columns both have high number of diversity and long values in rows. STR6 is separated from these group by not having any missing value due to the direct relationship with the target column.

4.2 Text Processing Model Building

Like the previous sections of this study KNIME is used to develop models on the dataset. Six parallel models are built for each text column and two extra model is built for the combination of those columns. These two differs from each other for their preprocessing sections. For text processing applications punctuation, number, short words stop words or tag filtering are applied on data as can be seen at Figure 4.2.1. For achieve a proper comparison and see the effects of these filters, one model is developed without these preprocessing steps.

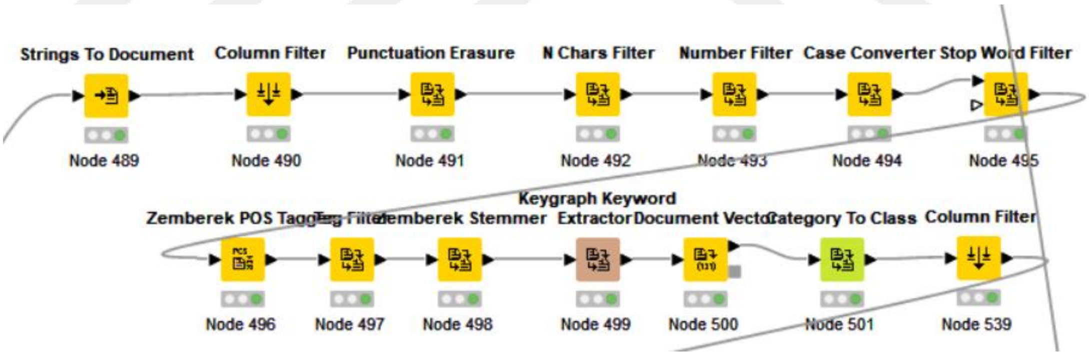


Figure 4.2.1 Preprocessing and classification part of text processing workflow

KNIME text processing extension uses data as documents; therefore, all text data is transformed into document data with the first node of the preprocessing step. Then the data cleansing part starts in the workflow. Alphanumeric characters are removed in first cleansing step. The next step is removing the characters that have less than a specified number of characters. In this scenario the number is set to two. Afterwords, numeric characters are filtered, and all string values are set to same, lowercase. Meaning analysis is started after character elimination process. Stop words that are meaningless like connectors in sentences are eliminated. Elimination process ending leads to the key word extraction part of the text processing model. KNIME has a special node to use in Turkish language. Selected key words with these nodes are turned into a vector and vector is transformed to binary classes. With all these implementations, text data becomes useful for a mathematical model. The new data extracted from the combined six columns for example, have 404 columns and the same row amount of the original data. The rest of the model consists of the Decision Tree Learner, Predictor and Scorer nodes of KNIME for all options that is analyzed.

4.3 Text Processing Model Comparison

As mentioned at the previous section, eight models developed in KNIME for comparison as columns by themselves and combination of textual data in one column. Each model used Decision Trees as classification algorithm and the scores of them are obtained and showed in Table 4.3.1. Only textual data and the target column are used as input data in these models. The results of the individual columns vary from each other in all metrics. From that data that is shown in tab it can be said that STR5 seams quite accurate but have very less data point which can be resulted from having the highest number of missing values. STR6 seems much more effective on the decision process than STR5. It also provides much more data point. Moreover, when we look back the number of missing information about STR6, which is zero, it supports the performance results. On the other hand, STR1, STR2 and STR4 columns have relatively worse results at measurement metrics. This is expected due to the lack of total data points.

Table 4.3.1 Metrics for text processing models

Model/ Metrics	STR1	STR2	STR3	STR4	STR5	STR6	Combined	Combined with no filter
Correct	26429	29796	38009	6019	432	86523	54630	22311
Wrong	17886	12639	3694	2639	29	1912	3800	1286
Accuracy	0.60	0.70	0.91	0.70	0.94	0.98	0.94	0.95
Error	0.40	0.30	0.09	0.30	0.06	0.02	0.06	0.05
Cohen’s Kappa	0.30	0.18	0.87	0.00	0.05	0.93	0.89	0.91

Combination of text columns has results close to the full column STR6 but below them. This leads to the questioning the necessity of the other columns because only STR6 has a data for each row of target column. Missing value effect can be observed from these results. Preprocessing of text data makes data set volume in the column direction so much higher and does not bring that much effect on the metric results. From that point of view preprocessed combined text is the most appropriate to use for a hybrid model.

4.4 Combined Machine Learning Model

A combination model from all data can be built with both using text processing and classification algorithms. This proposed model begins with the same preprocessing steps used in classification model developed in the first part of the study. Following that text processing part is added to the model for the combined string column obtained from the textual data studied in the second part of this study. After that, all data joined in one singular table, which has become a table with 3971 columns, for implementation of machine learning algorithms. This preprocessing part model can be seen at the Figure 4.4.1.

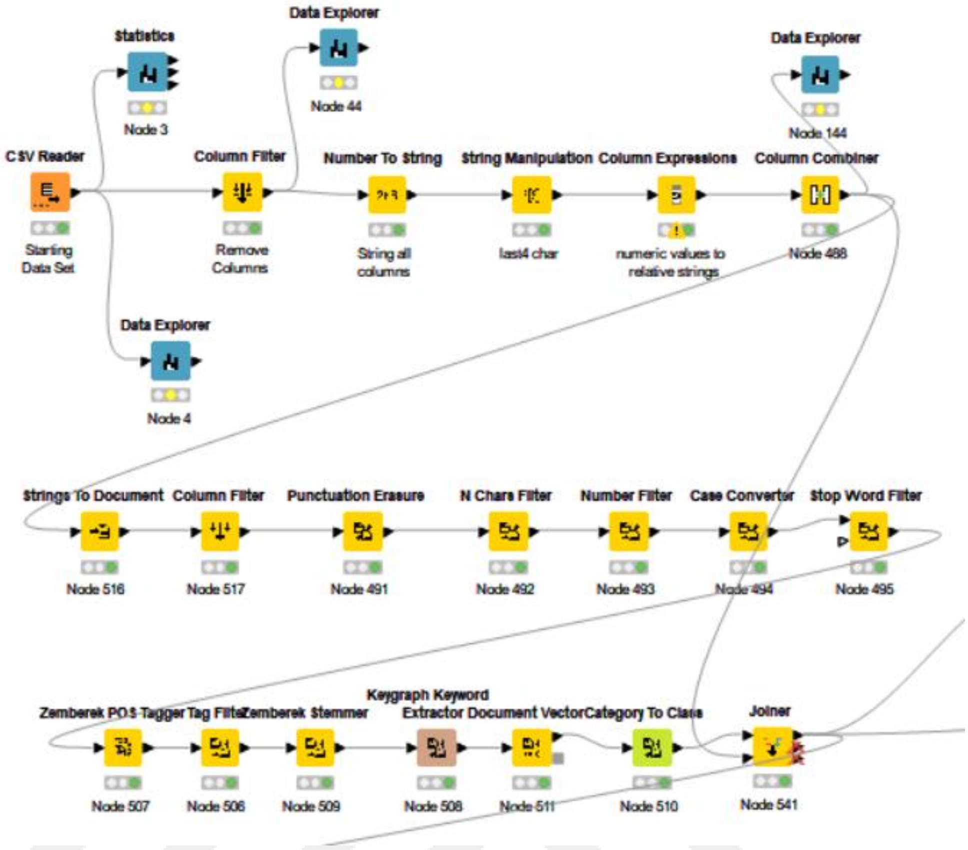
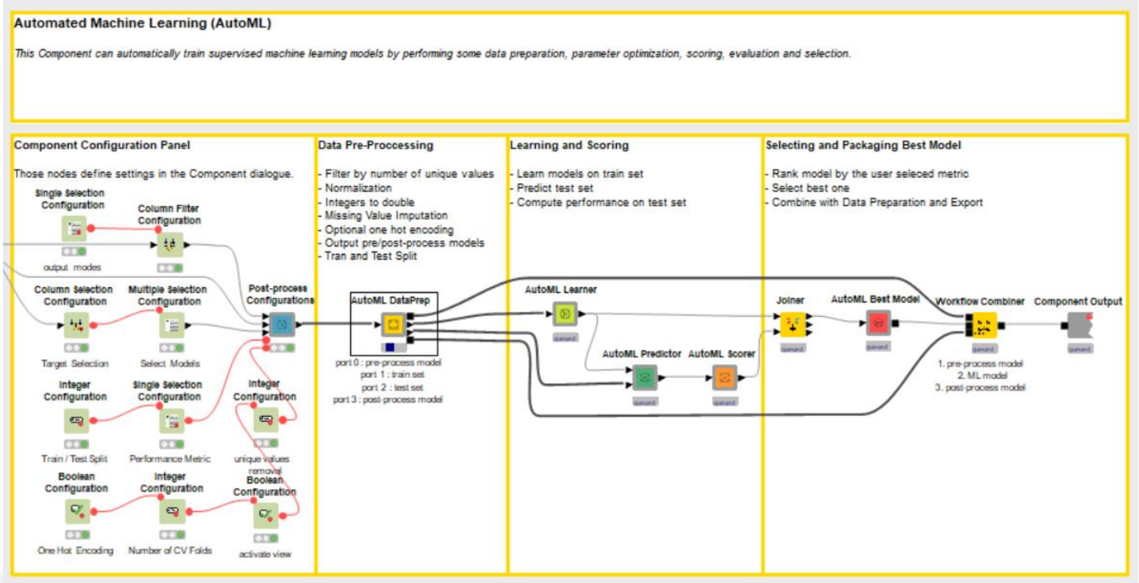


Figure 4.4.1 Preprocessing of data in proposed combined model

Machine learning model part of this combination is decided by using automl again. Preprocessed data is used as an input in automl to observe the best possible model. That model is applied in feature selection to select the most effective features in data set. Automl makes parameter optimization and more preprocessing on data. The sections of this component can be seen in figure 4.4.2. Those properties and data modifications also done on the final workflow. The resulted features and model are selected for the final machine learning workflow and this process can be repeatedly applied with the increased or enhanced data in the future to improve the final model.



Automated Machine Learning (AutoML)

This Component can automatically train supervised machine learning models by perfoming some data preparation, parameter optimization, scoring, evaluation and selection.

Component Configuration Panel

Those nodes define settings in the Component dialogue.

Data Pre-Proccessing

- Filter by number of unique values
- Normalization
- Integers to double
- Missing Value Imputation
- Optional one hot encoding
- Output pre/post-process models
- Tran and Test Split

Learning and Scoring

- Learn models on train set
- Predict test set
- Compute performance on test set

Selecting and Packaging Best Model

- Rank model by the user seleced metric
- Select best one
- Combine with Data Preparation and Export

Figure 4.4.2 KNIME AutoML main sections

The approach consists of applying 10 models on same dataset, one machine learning algorithm is suggested by automl process. This same algorithm is used in feature selection loop and the most usefull features of data is selected with this process. At the end, same algorithm is applied to selected features for classifying the target column. This process can be repeated in selected time periods to justify the model an make improvements if it is needed.

5. DISCUSSION

This study aims to compare the knowledge-based data effect on a regular decision-making model. Technical structured data and textual data of defect decision are analyzed in this study. Technical data analyzed with classification algorithms and the metric results are taken. Following that, the existence of text data is added as an information into the models. The effect of this addition to the results are observed. With classification algorithms approximately 30% error is retrieved and another metric, cohen’s kappa is around 0.3. With the addition of the text columns, both measurement metrics improve regardless of the data itself. However, the best model still has 24% error rate and 0.6 cohen’s kappa value. With parameter tunning applications and alternative algorithm applications these metrics can be improved. This analysis seeks an answer for the first research question of this study.

Textual data by itself is studied in the second part of this study in order to develop a solution for the second research question of this study. Both individual columns and combined text data is analyzed in terms of statistical metrics. Preprocessing of data is tested in combined version of data also. Column data quality effect on the result can be easily observed. High quality and availability of a textual column show the best result in text processing. On the other hand, combined text does not bring metric measurements as good as one specific column that has information about every row for target column.

With those analysis, it can be observed that for defect decision processes knowledge data is a very powerful resource. In literature, there are sources supporting this conclusion by proposing hybrid decision support systems or decision-making model that concerns knowledge data. Sharma and Kumar (2021, 8) explain the advantages of the implementation of a knowledge-based decision support system despite of being hard to implement. Psarommatis and Kiritsis (2022, 13) proposed a hybrid decision support system for defect evaluation and achieved approximately 7% increase in performance. Those examples from literature strengthens the idea of using knowledge data in defect evaluation processes.

Problem defined by partner company is deeply analyzed with this study and resulted as the textual data is the major input for defect evaluation. Data provided by the responsible department is tested and the effects on the results are analyzed. With the help of the study itself and the literature resources, tested models and preprocessed data are proposed in another model as a possible answer for the final research question of this study. At this model, both feature selection property of KNIME and parameter optimization property of automl combined. The suggested model will be tested in following projects that is planned by the company. Also, the data used in this paper should be reconsidered and further additions of knowledge data should be investigated. This paper showed that textual data can be used as an input for machine learning application in manufacturing environment like marketing or customer relations departments.

6. CONCLUSION

The main contribution of this study is to demonstrate the impact of knowledge-based data on a regular decision-making model. The results show that using text columns as additional features can improve the performance of classification algorithms and reduce the error rate. Furthermore, the study proposes a novel approach for extracting and encoding relevant information from text documents using natural language processing techniques. The approach can be applied to various domains and scenarios where textual data is available and important for decision making.

The data used can be subject to the main limitation for this study. Since the textual data for the defect or process of the defect decision is a newly adapted concept for the company, the completeness of the data is an issue. Therefore, it is hard to generalize the results from this analysis. The other critical limitation is about the computation resource used in this study. It restricted the variety of the algorithms that are executed for the dataset.

Despite those limitations, the contributions of this study can be associated with the key research questions mentioned in the introduction section. Historical defect decision data can be employed to support the decision-making process about defect with some restrictions. The defect decision is a multiclass data and the classification of that data is naturally a daunting problem. The data used in this study presents some insights for a decision, but it should be enhanced for more effective results. That enhancement need can be partially met by adding the textual data. Textual data can be transformed into a suitable data frame for a classification algorithm by text mining applications. Therefore, those two kinds of data can be combined in a one machine learning model that utilizes both classification and text mining techniques for this multiclass classification problem that this study investigates.

7. FUTURE WORKS

The proposed model is a pre investigation for an R&D project at the partner company and because of that, future works are already planned. Testing other models on data that is not mentioned in this study, combination of the two aspects of the model, parameter optimization works and transforming into an executable software during process are the beginning of this plan. Enriching data with different corporate data and analyze the results is another step that is triggered with this study. Not only quality data, but also order or customer related data should be considered in this analysis. Also, besides the notes, document-based data and image data are eligible to use in this model. Therefore, model should be revised in order to comply to these extensions. All these future works will be planned and run with R&D department of the partner company and will be proposed to the governmental officials with essential reports and papers.

BIBLIOGRAPHY

Borcelik n.d. “Corporate: About Us” Accessed March 8, 2024. https://www.borcelik.com/en/corporate/about-us.

Breiman, Leo. 2001. "Random Forests." Machine Learning 45, no. 1: 5-32. https://doi.org/10.1023/A:1010933404324.

Chen, Zheyuan. 2022. "Data Mining for Fault Diagnosis in Steel-making Process under Industry 4.0." PhD Thesis, Cardiff University.

Cortes, Corinna, and Vladimir Vapnik. 1995. "Support-Vector Networks." Machine Learning 20, no. 3: 273-297. https://doi.org/10.1007/BF00994018.

World Steel Association n.d “Media: Press Releases: 2024: December 2023 crude steel production and 2023 global crude steel production totals." Accessed July 22, 2024. https://worldsteel.org/media/press-releases/2024/december-2023-crude-steel-production-and-2023-global-totals/.

Feldman, Ronen, and James Sanger. 2007. The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press. https://www.researchgate.net/publication/200504395_The_text_mining_handbook_Advanced_approaches_in_analyzing_unstructured_data

Hao, R., Lu, B., Cheng, Y., Li, X., & Huang, B. 2021. “A steel surface defect inspection approach towards smart industrial monitoring.” *Journal of Intelligent Manufacturing*, 32, 1833-1843. https://doi.org/10.1007/s10845-020-01670-2.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. Springer. https://hastie.su.domains/ElemStatLearn/

International Organization for Standardization. "Quality management systems - Fundamentals and vocabulary." ISO 9000:2015. Accessed January 1, 2022. https://www.iso.org/standard/62085.html.

J.Mohammad, Mohammad Alkin. 2007. “Text Mining : A Burgeroning Quality Improvement Tool.” M.S. - Master of Science, Middle East Technical University.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. An Introduction to Statistical Learning: With Applications in R. Springer. https://www.statlearning.com/

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. "Deep Learning." *Nature*

521 (7553): 436–44. https://doi.org/10.1038/nature14539.

Magar, V. M., and V. B. Shinde. 2014. "Application of 7 quality control (7 QC) tools for continuous improvement of manufacturing processes." *International Journal of Engineering Research and General Science* 2, no. 4: 364-371. https://api.semanticscholar.org/CorpusID:111148751.

Mitchell, Tom M. 1997. Machine Learning. McGraw-Hill. https://www.cs.cmu.edu/~tom/mlbook.html

US Assistant Secretary of Defense. 1965. Guide To Zero Defects. Manpower Installations and Logistics. Qual. Reliab. Assur. Handbook,Washington, DC. https://apps.dtic.mil/sti/tr/pdf/ADA950061.pdf

Preeti. 2021. "Review on Text Mining: Techniques, Applications and Issues." In *2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART)*, 474–78. https://doi.org/10.1109/smart52563.2021.9676285.

Psarommatis, Foivos, and Dimitris Kiritsis. 2022. "A Hybrid Decision Support System for Automating Decision Making in the Event of Defects in the Era of Zero Defect Manufacturing." *Journal of Industrial Information Integration* 26: 100263. https://doi.org/10.1016/j.jii.2021.100263.

Psarommatis, Foivos, Gökan May, Paul-Arthur Dreyfus, and Dimitris Kiritsis. 2019. “Zero Defect Manufacturing: State-of-the-Art Review, Shortcomings and Future Directions in Research.” *International Journal of Production Research* 58 (1): 1–17. doi:10.1080/00207543.2019.1605228.

Sharma, Deepak, and Sunil Kumar. 2021. "Knowledge Based Decision Support System in Steel Industries." *IOP Conference Series: Materials Science and Engineering*. 1116: 012083. https://doi.org/10.1088/1757-899X/1116/1/012083.

Soares J.C, Tereso A.P., Sousa S.D. 2019. "A Decision-Making Model for the Rework of Defective Products." *International Journal of Quality & Reliability Management*, vol. 38, no. 1. 68-97. https://doi.org/10.1108/IJQRM-06-2019-0185.

Talib, Ramzan, Muhammad Kashif, Shaeela Ayesha, and Fakeeha Fatima. 2016. "Text Mining: Techniques, Applications and Issues." *International Journal of Advanced Computer Science and Applications* 7. doi:10.14569/IJACSA.2016.071153.

Tiensuu, Henna, Satu Tamminen, Esa Puukko, and Juha Röning. 2021. "Evidence-Based and Explainable Smart Decision Support for Quality Improvement in Stainless Steel Manufacturing" *Applied Sciences* 11, no. 22: 10897. https://doi.org/10.3390/app112210897

Wang, Yao, Zhaoyun Zhang, Zheng Wang, Cheng Wang, and Cheng Wu. “Interpretable Machine Learning-Based Text Classification Method for Construction Quality Defect Reports.” *Journal of Building Engineering* 89 (July 2024): 109330. https://doi.org/10.1016/j.jobe.2024.109330.

Wang, Zhaoping, Jian Wang, and Sen Chen. 2020. “Fault Location of Strip Steel Surface Quality Defects on Hot-Rolling Production Line Based on Information Fusion of Historical Cases and Process Data.” *IEEE Access* 8: 171240–51. https://doi.org/10.1109/access.2020.3024582.

Wen, Xin, Jvran Shan, Yu He, and Kechen Song. 2023. "Steel Surface Defect Recognition: A Survey" Coatings 13, no. 1: 17. https://doi.org/10.3390/coatings13010017

Zhang, Y., H. Wang, W. Shen, and G. Peng. 2023 "DuAK: Reinforcement Learning-Based Knowledge Graph Reasoning for Steel Surface Defect Detection." *IEEE Transactions on Automation Science and Engineering*, 1-13. doi:10.1109/TASE.2023.3307588.

APPENDIX A

A.1 Data Analysis for Classification

There are 62 string type columns in starting data set. Number of missing and unique values are given in table below.

Table A.1 String Columns information from the starting data set

Column	**No. missings**	**Unique values**
STR1	0	>1000
STR2	0	19
STR3	5	48
STR4	0	>1000
STR5	0	546
STR6	0	>1000
STR7	1457	286
STR8	1455	896
STR9	0	125
STR10	0	15
STR11	0	77
STR12	6	417
STR13	65031	999
STR14	212521	366
STR15	0	>1000
STR16	213024	999
STR17	1455	11
STR18	390148	324
STR19	389448	5
STR20	19	>1000
STR21	0	>1000
STR22	43	530
STR23	6	231
STR24	3675	24
STR25	1455	5
STR26	10	72
STR27	0	>1000
STR28	41612	>1000
STR29	0	>1000
STR30	0	1
STR31	300838	>1000
STR32	173718	3
STR33	389890	42
STR34	301933	494

STR35	67077	700
STR36	67077	245
STR37	393366	2
STR38	393393	0
STR39	393393	0
STR40	393393	0
STR41	393393	0
STR42	393393	0
STR43	393391	1
STR44	393393	0
STR45	393393	0
STR46	1448	62
STR47	44	840
STR48	225773	11
STR49	1455	3
STR50	388937	787
STR51	800	24
STR52	2219	16
STR53	385291	8
STR54	385304	12
STR55	393377	6
STR56	113884	10
STR57	48282	10
STR58	386640	1
STR59	10798	7
STR60	343	>1000
STR61	375211	1
STR62	393393	0

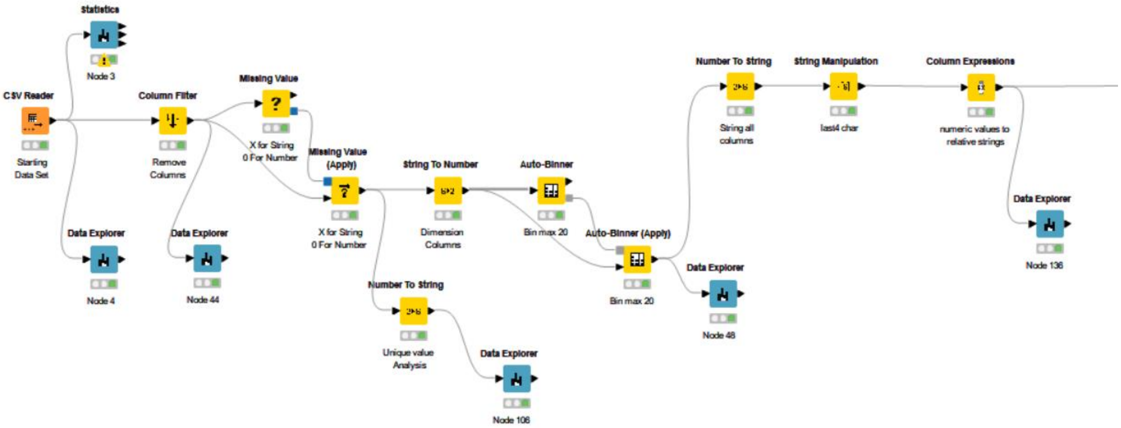
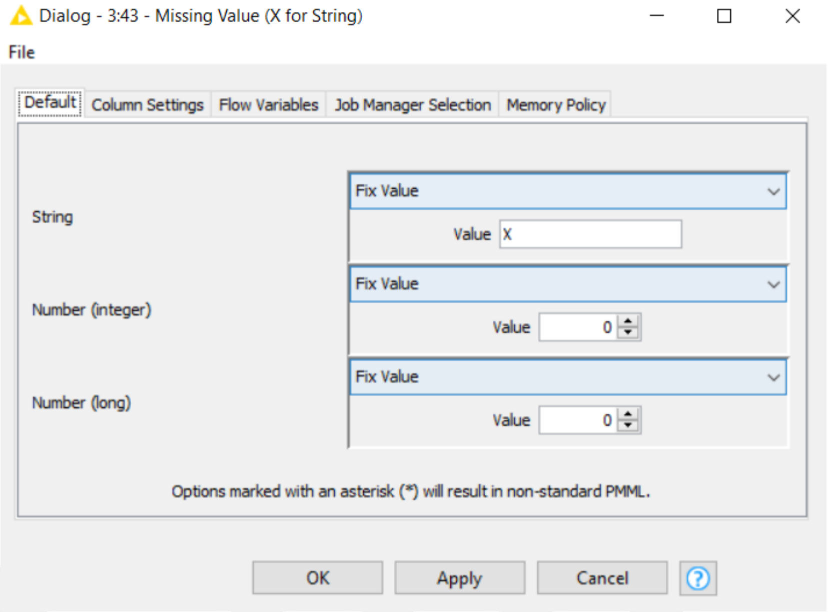


Figure A.1 KNIME Workflow for data preparation



Dialog - 3:43 - Missing Value (X for String)

File

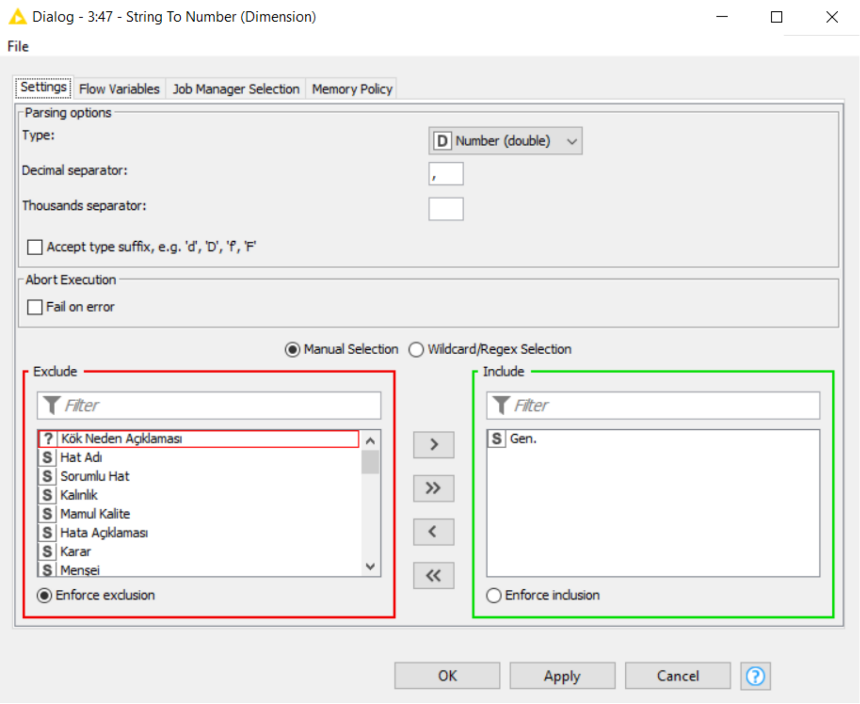
String: Fix Value — Value: X

Number (integer): Fix Value — Value: 0

Number (long): Fix Value — Value: 0

Options marked with an asterisk (*) will result in non-standard PMML.

Figure A.2 KNIME Missing Value Node



Dialog - 3:47 - String To Number (Dimension)

File

Parsing options

Type: D Number (double)

Decimal separator: ,

Thousands separator:

□ Accept type suffix, e.g. 'd', 'D', 'f', 'F'

Abort Execution

□ Fail on error

● Manual Selection ○ Wildcard/Regex Selection

Exclude

? Kök Neden Açklaması
S Hat Adı
S Sorumlu Hat
S Kalınlık
S Mamul Kalite
S Hata Açıklaması
S Karar
S Menşei

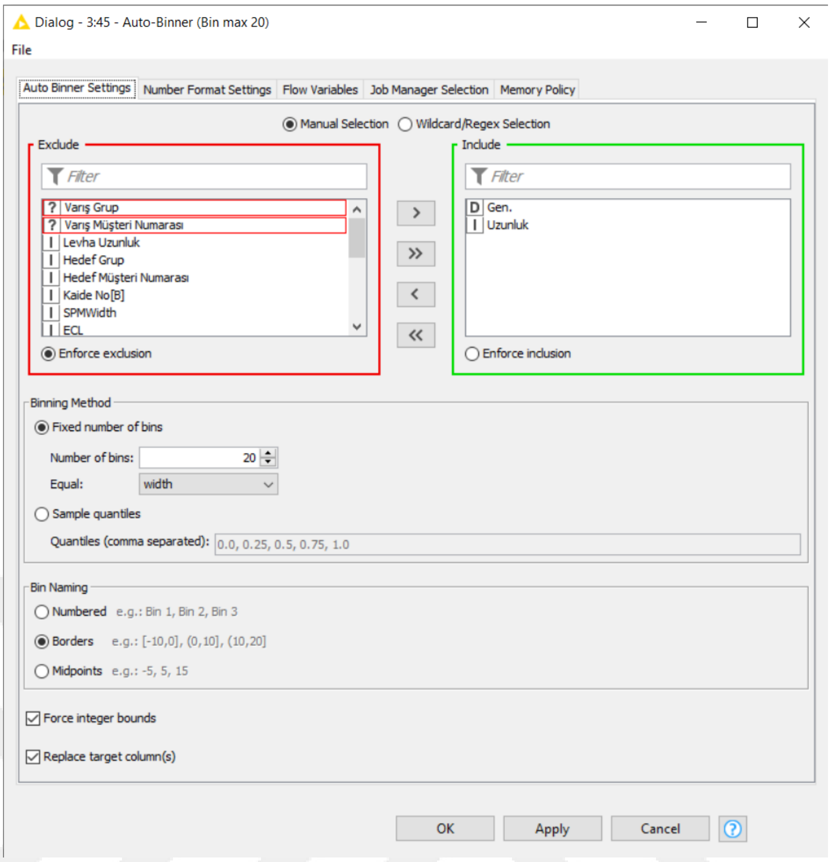
● Enforce exclusion

Include

S Gen.

○ Enforce inclusion

Figure A.3 KNIME String To Number Node



● Manual Selection ○ Wildcard/Regex Selection

Exclude

- Varış Grup
- Varış Müşteri Numarası
- Levha Uzunluk
- Hedef Grup
- Hedef Müşteri Numarası
- Kaide No[B]
- SPMWidth
- ECL

● Enforce exclusion

Include

- Gen.
- Uzunluk

○ Enforce inclusion

Binning Method

● Fixed number of bins

Number of bins: 20

Equal: width

○ Sample quantiles

Quantiles (comma separated): 0.0, 0.25, 0.5, 0.75, 1.0

Bin Naming

○ Numbered e.g.: Bin 1, Bin 2, Bin 3

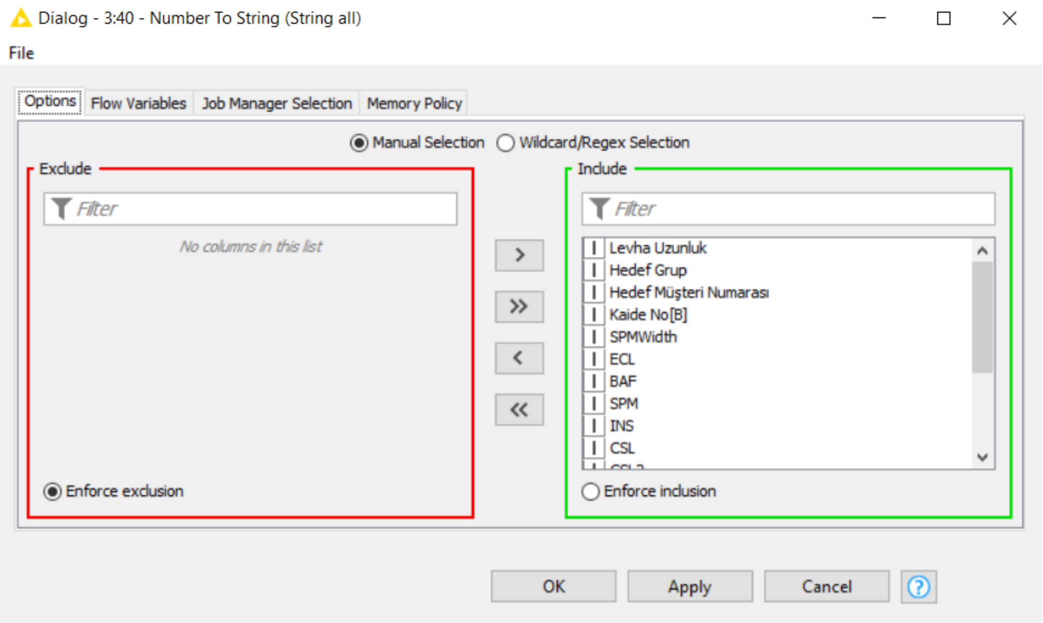
● Borders e.g.: [-10,0], (0,10], (10,20]

○ Midpoints e.g.: -5, 5, 15

☑ Force integer bounds

☑ Replace target column(s)

Figure A.4 KNIME Auto Binner Node



● Manual Selection ○ Wildcard/Regex Selection

Exclude

No columns in this list

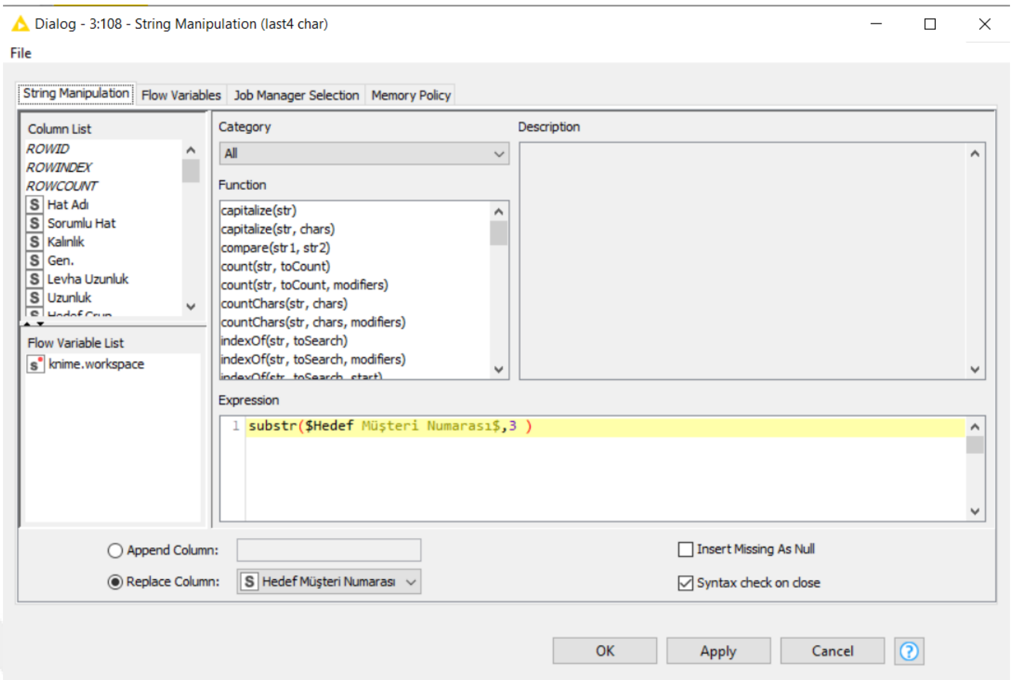
● Enforce exclusion

Include

- Levha Uzunluk
- Hedef Grup
- Hedef Müşteri Numarası
- Kaide No[B]
- SPMWidth
- ECL
- BAF
- SPM
- INS
- CSL

○ Enforce inclusion

Figure A.5 KNIME Number To String Node



Column List
ROWID
ROWINDEX
ROWCOUNT
S Hat Adı
S Sorumlu Hat
S Kalınlık
S Gen.
S Levha Uzunluk
S Uzunluk

Flow Variable List
s knime.workspace

Category: All

Function
capitalize(str)
capitalize(str, chars)
compare(str1, str2)
count(str, toCount)
count(str, toCount, modifiers)
countChars(str, chars)
countChars(str, chars, modifiers)
indexOf(str, toSearch)
indexOf(str, toSearch, modifiers)

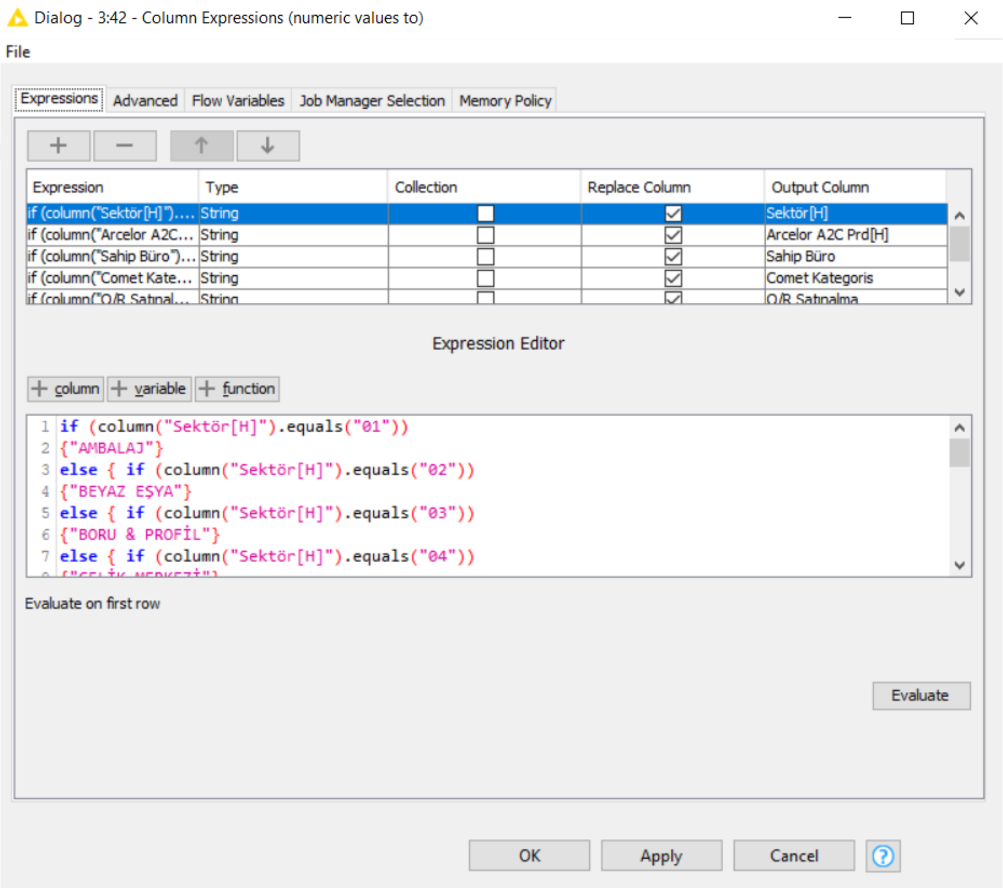
Description

Expression
```
substr($Hedef Müşteri Numarası$,3 )
```

○ Append Column:
● Replace Column: S Hedef Müşteri Numarası

□ Insert Missing As Null
☑ Syntax check on close

Figure A.6 KNIME String Manipulation Node



Expression	Type	Collection	Replace Column	Output Column
if (column("Sektör[H]")….	String	□	☑	Sektör[H]
if (column("Arcelor A2C…	String	□	☑	Arcelor A2C Prd[H]
if (column("Sahip Büro")…	String	□	☑	Sahip Büro
if (column("Comet Kate…	String	□	☑	Comet Kategoris
if (column("O/R Satınal…	String	□	☑	O/R Satınalma

Expression Editor

```
if (column("Sektör[H]").equals("01"))
{"AMBALAJ"}
else { if (column("Sektör[H]").equals("02"))
{"BEYAZ EŞYA"}
else { if (column("Sektör[H]").equals("03"))
{"BORU & PROFİL"}
else { if (column("Sektör[H]").equals("04"))
```

Evaluate on first row

Figure A.7 KNIME Column Expression Node

A.2 Model Building for Classification

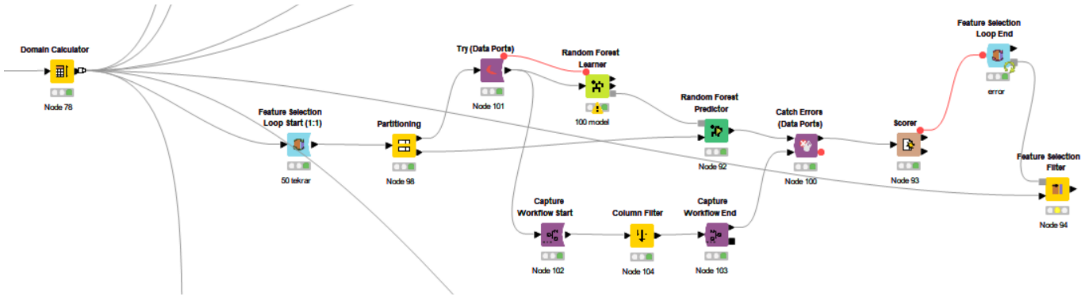
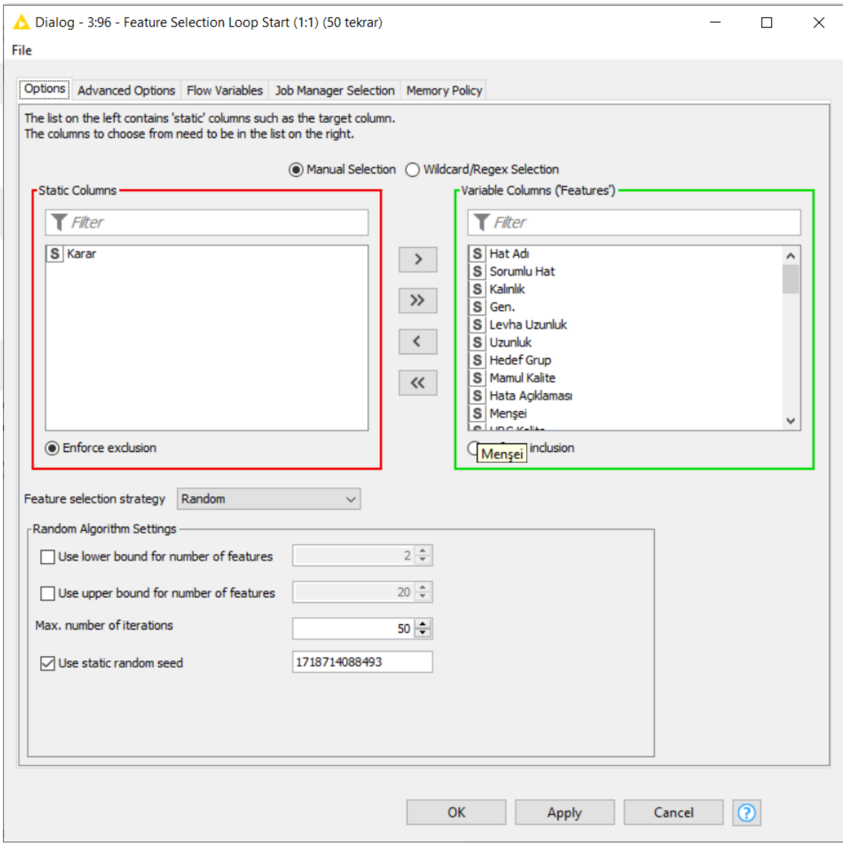


Figure A.8 KNIME Workflow for classification models



The list on the left contains 'static' columns such as the target column.
The columns to choose from need to be in the list on the right.

● Manual Selection ○ Wildcard/Regex Selection

Static Columns

S Karar

● Enforce exclusion

Variable Columns ('Features')

S Hat Adı
S Sorumlu Hat
S Kalınlık
S Gen.
S Levha Uzunluk
S Uzunluk
S Hedef Grup
S Mamul Kalite
S Hata Açıklaması
S Menşei

○ Menşei inclusion

Feature selection strategy: Random

Random Algorithm Settings

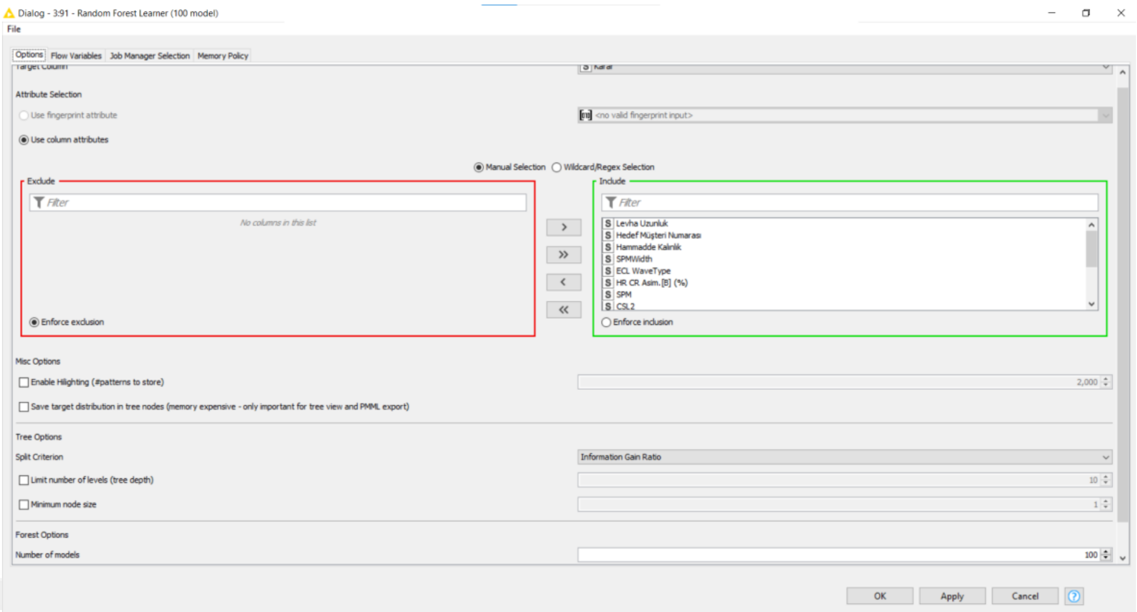
☐ Use lower bound for number of features: 2

☐ Use upper bound for number of features: 20

Max. number of iterations: 50

☑ Use static random seed: 1718714088493

Figure A.9 KNIME Feature Selection Loop Start Node



Dialog - 3:91 - Random Forest Learner (100 model)

Target Column: S Karar

Attribute Selection

○ Use fingerprint attribute <no valid fingerprint input>

● Use column attributes

● Manual Selection ○ Wildcard/Regex Selection

Exclude

No columns in this list

● Enforce exclusion

Include

- S Levha Uzunluk
- S Hedef Müşteri Numarası
- S Hammadde Kalınlık
- S SPMWidth
- S ECL WaveType
- S HR CR Asim.[B] (%)
- S SPM
- S CSL2

○ Enforce inclusion

Misc Options

□ Enable Hilighting (#patterns to store) 2,000

□ Save target distribution in tree nodes (memory expensive - only important for tree view and PMML export)

Tree Options

Split Criterion: Information Gain Ratio

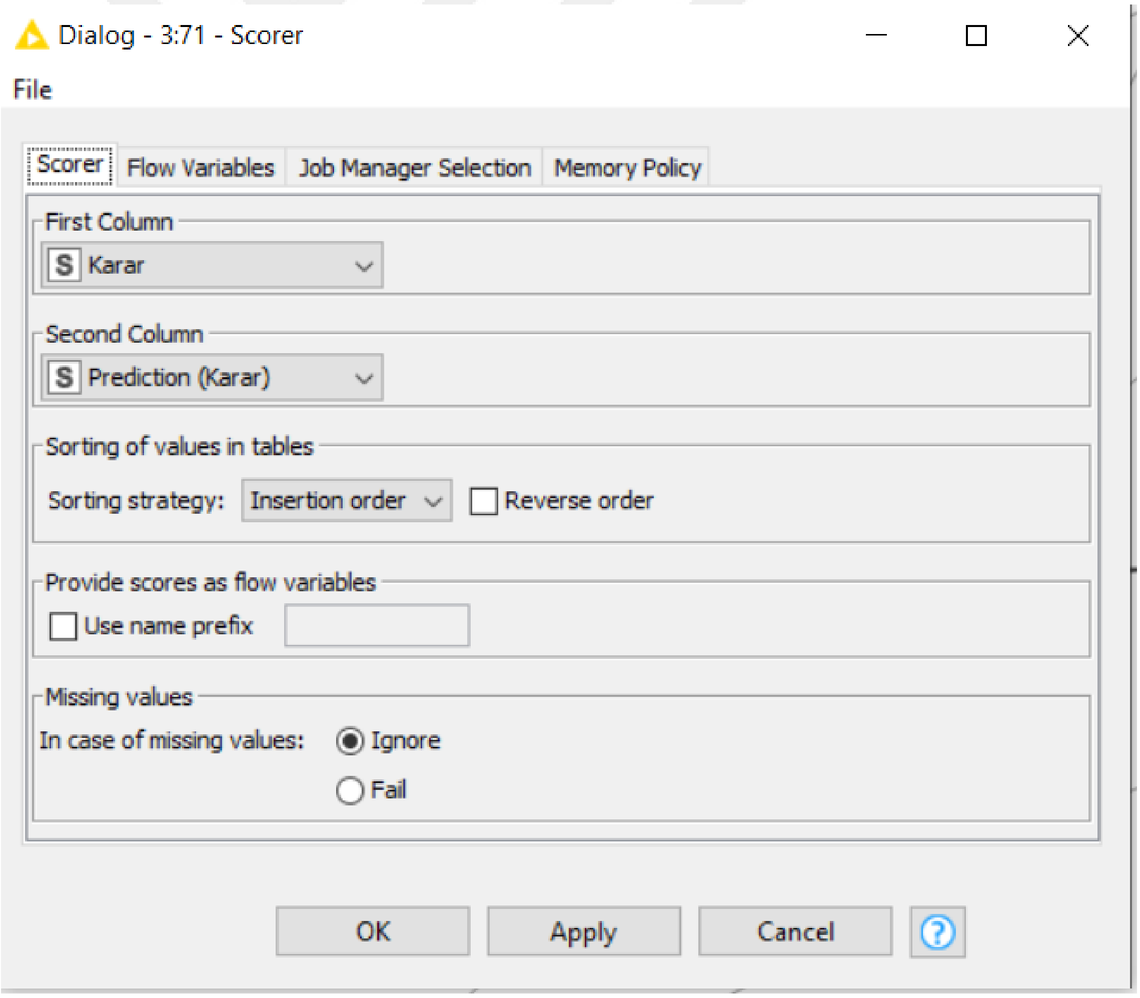
□ Limit number of levels (tree depth) 10

□ Minimum node size 1

Forest Options

Number of models: 100

Figure A.10 KNIME Random Forest Learner Node



Dialog - 3:71 - Scorer

First Column: S Karar

Second Column: S Prediction (Karar)

Sorting of values in tables

Sorting strategy: Insertion order □ Reverse order

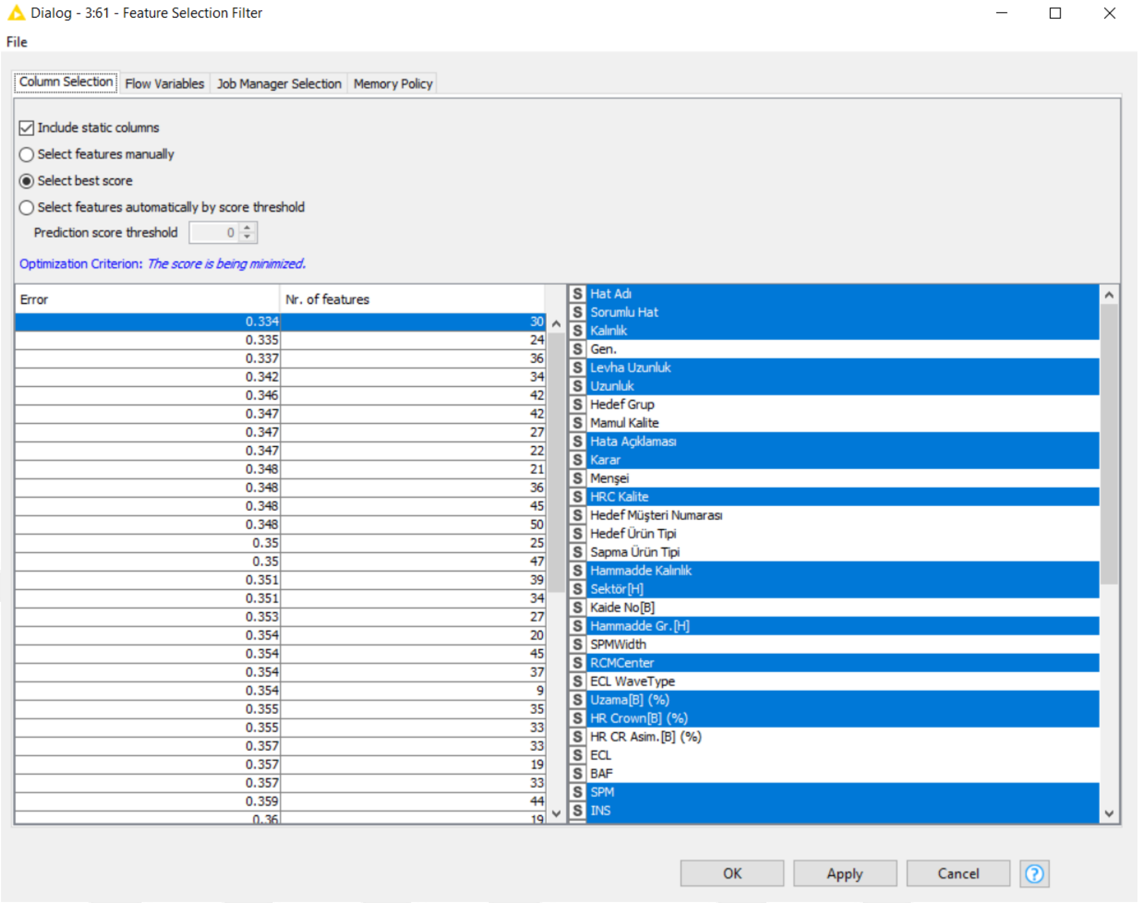
Provide scores as flow variables

□ Use name prefix

Missing values

In case of missing values: ● Ignore ○ Fail

Figure A.11 KNIME Scorer Node



☑ Include static columns

○ Select features manually

● Select best score

○ Select features automatically by score threshold

Prediction score threshold 0

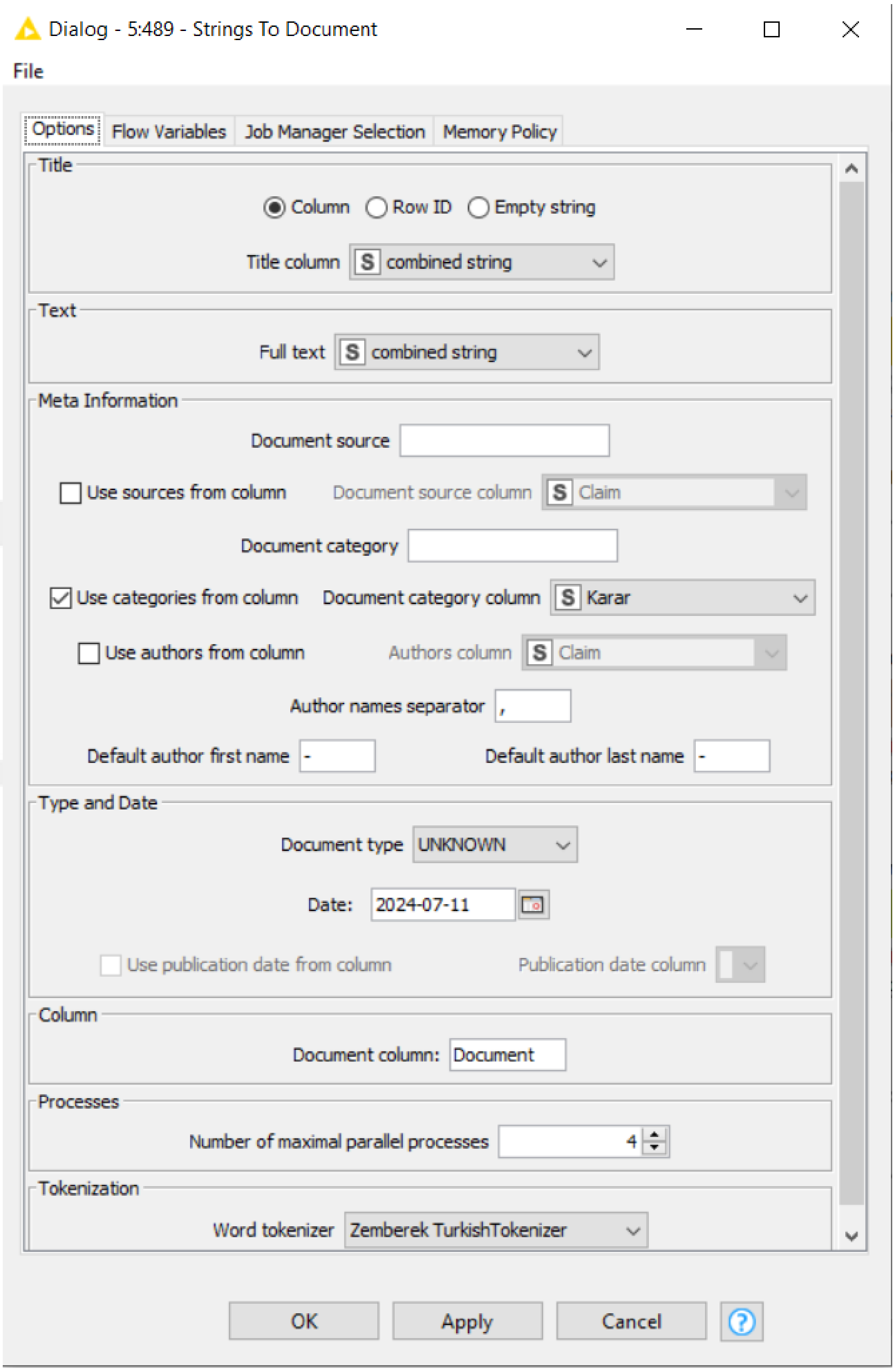
Optimization Criterion: *The score is being minimized.*

Error	Nr. of features
0.334	30
0.335	24
0.337	36
0.342	34
0.346	42
0.347	42
0.347	27
0.347	22
0.348	21
0.348	36
0.348	45
0.348	50
0.35	25
0.35	47
0.351	39
0.351	34
0.353	27
0.354	20
0.354	45
0.354	37
0.354	9
0.355	35
0.355	33
0.357	33
0.357	19
0.357	33
0.359	44
0.36	19

S Hat Adı
S Sorumlu Hat
S Kalınlık
S Gen.
S Levha Uzunluk
S Uzunluk
S Hedef Grup
S Mamul Kalite
S Hata Açıklaması
S Karar
S Menşei
S HRC Kalite
S Hedef Müşteri Numarası
S Hedef Ürün Tipi
S Sapma Ürün Tipi
S Hammadde Kalınlık
S Sektör[H]
S Kaide No[B]
S Hammadde Gr.[H]
S SPMWidth
S RCMCenter
S ECL WaveType
S Uzama[B] (%)
S HR Crown[B] (%)
S HR CR Asim.[B] (%)
S ECL
S BAF
S SPM
S INS

Figure A.12 KNIME Feature Selection Filter Node

A.3 Model Building for Text Processing



Title

● Column ○ Row ID ○ Empty string

Title column: S combined string

Text

Full text: S combined string

Meta Information

Document source:

□ Use sources from column — Document source column: S Claim

Document category:

☑ Use categories from column — Document category column: S Karar

□ Use authors from column — Authors column: S Claim

Author names separator: ,

Default author first name: -

Default author last name: -

Type and Date

Document type: UNKNOWN

Date: 2024-07-11

□ Use publication date from column — Publication date column:

Column

Document column: Document

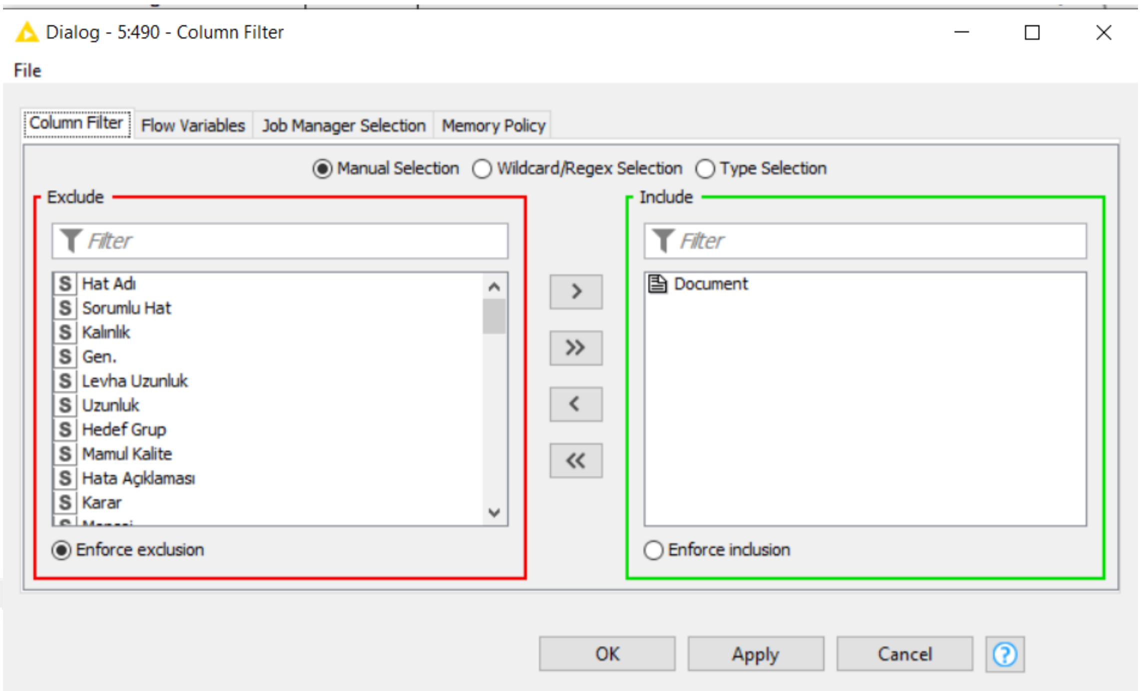
Processes

Number of maximal parallel processes: 4

Tokenization

Word tokenizer: Zemberek TurkishTokenizer

Figure A.13 KNIME Strings To Document Node


Dialog - 5:490 - Column Filter

File

Column Filter | Flow Variables | Job Manager Selection | Memory Policy

● Manual Selection ○ Wildcard/Regex Selection ○ Type Selection

Exclude

Filter

- S Hat Adı
- S Sorumlu Hat
- S Kalınlık
- S Gen.
- S Levha Uzunluk
- S Uzunluk
- S Hedef Grup
- S Mamul Kalite
- S Hata Açıklaması
- S Karar

● Enforce exclusion

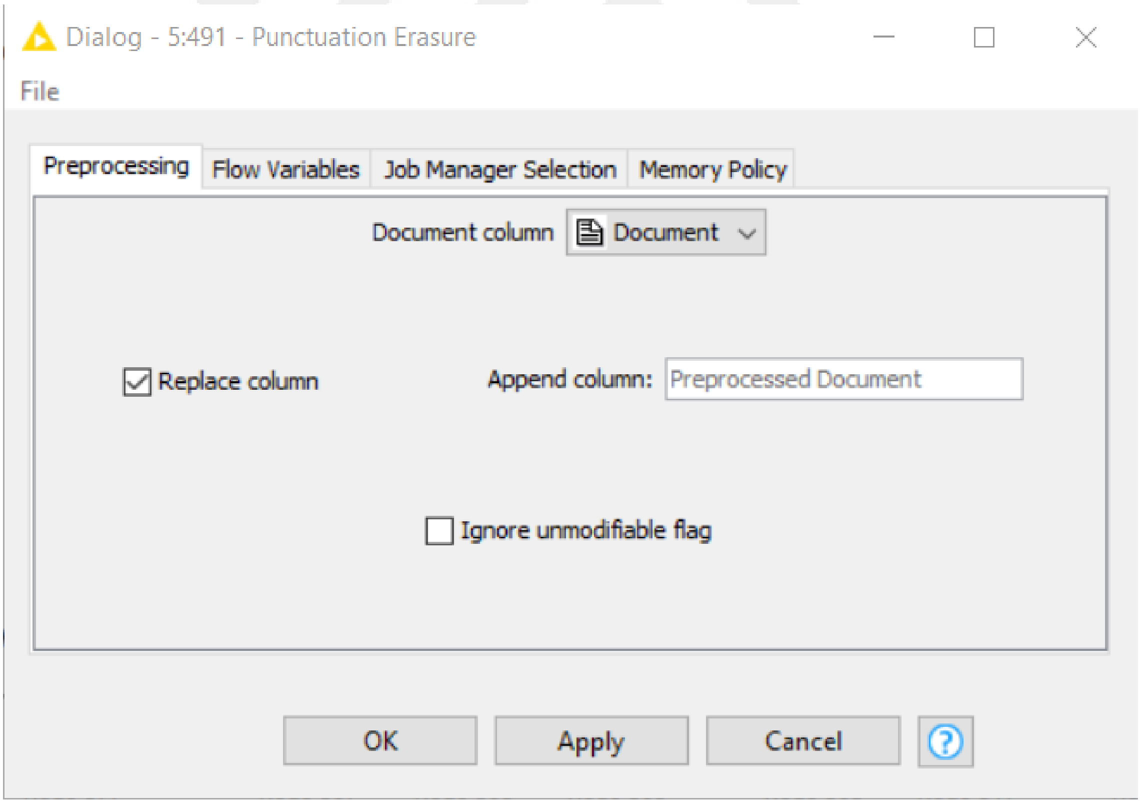
Include

Filter

- Document

○ Enforce inclusion

Figure A.14 KNIME Column Filter Node


Dialog - 5:491 - Punctuation Erasure

File

Preprocessing | Flow Variables | Job Manager Selection | Memory Policy

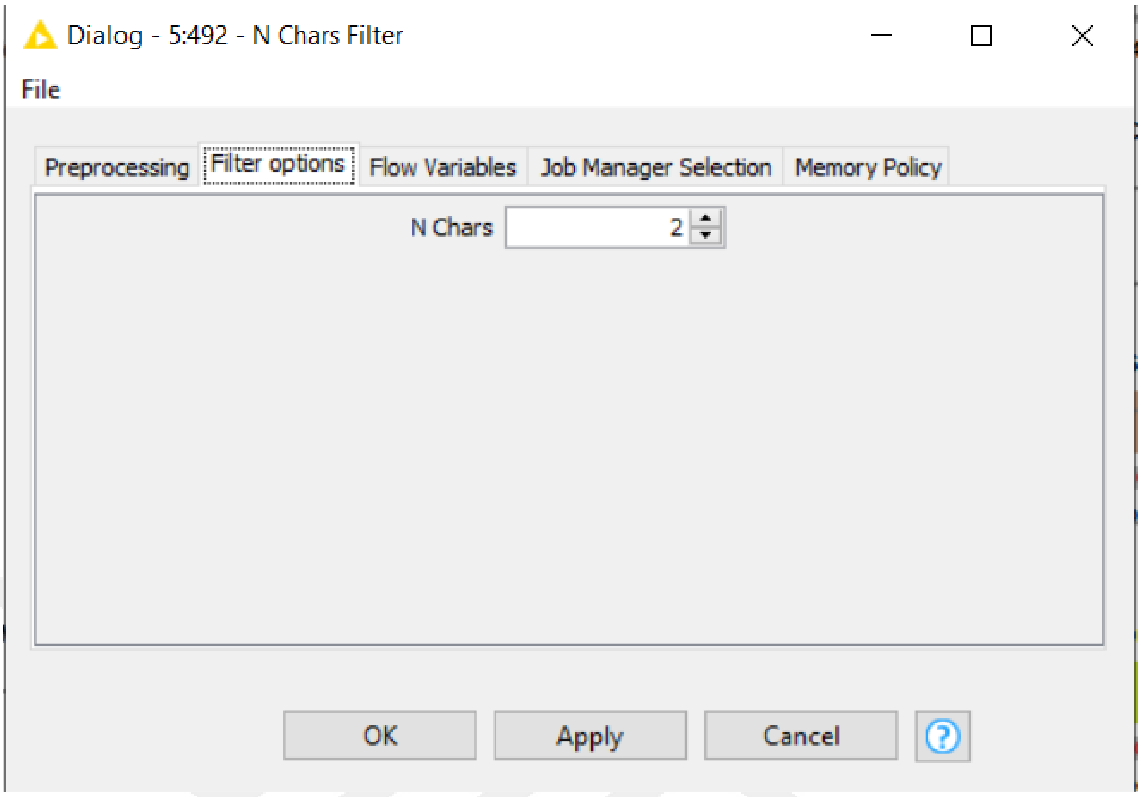
Document column: Document

☑ Replace column

Append column: Preprocessed Document

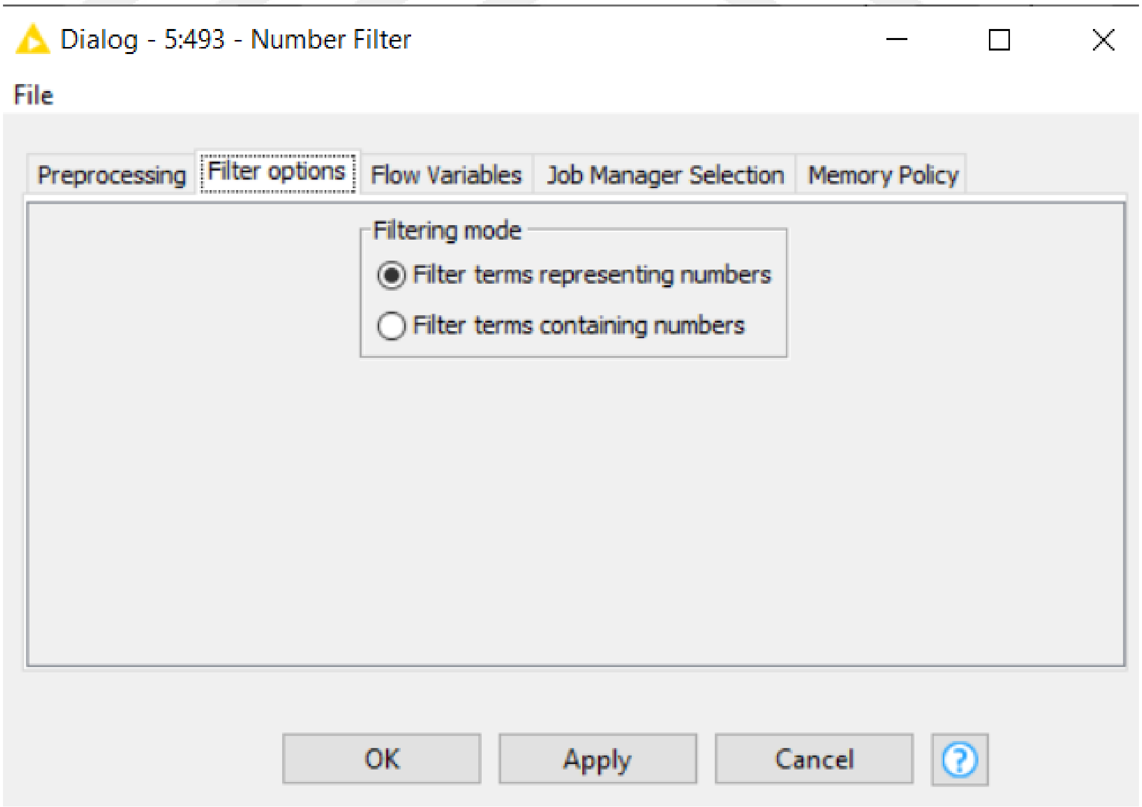
□ Ignore unmodifiable flag

Figure A.15 KNIME Punctuation Erasure Node



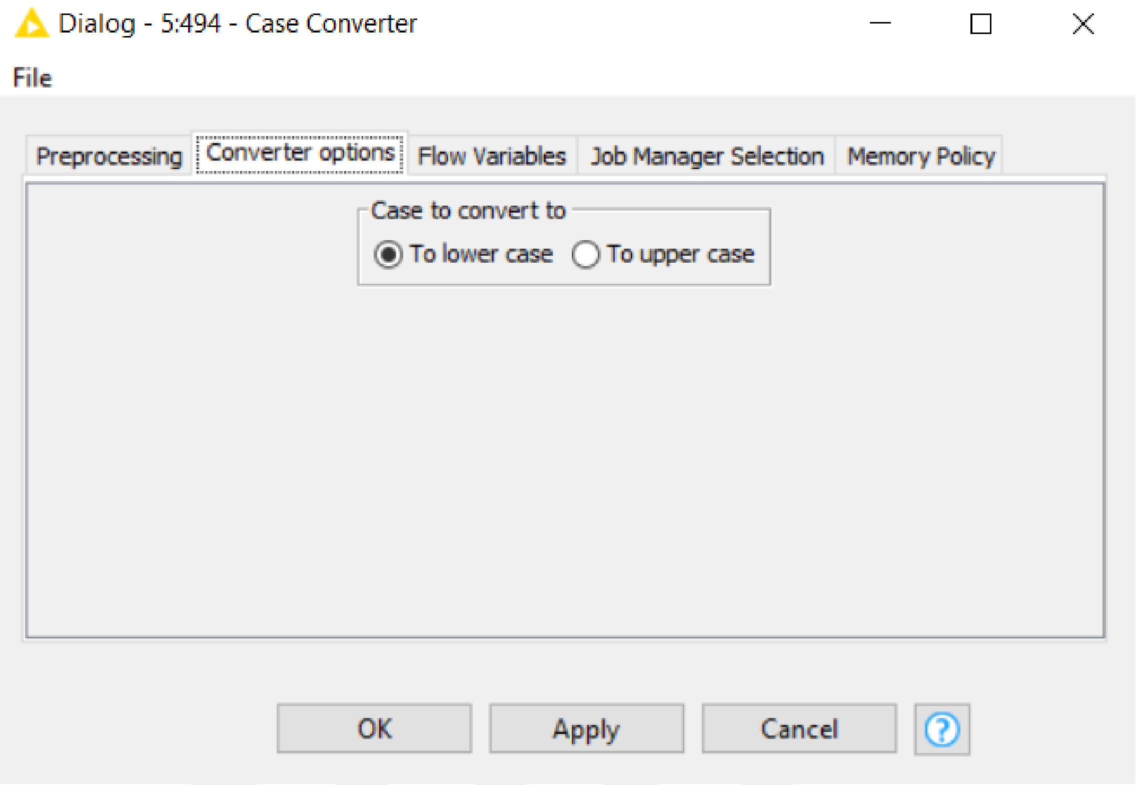
N Chars 2

Figure A.16 KNIME N Chars Filter Node



Filtering mode
● Filter terms representing numbers
○ Filter terms containing numbers

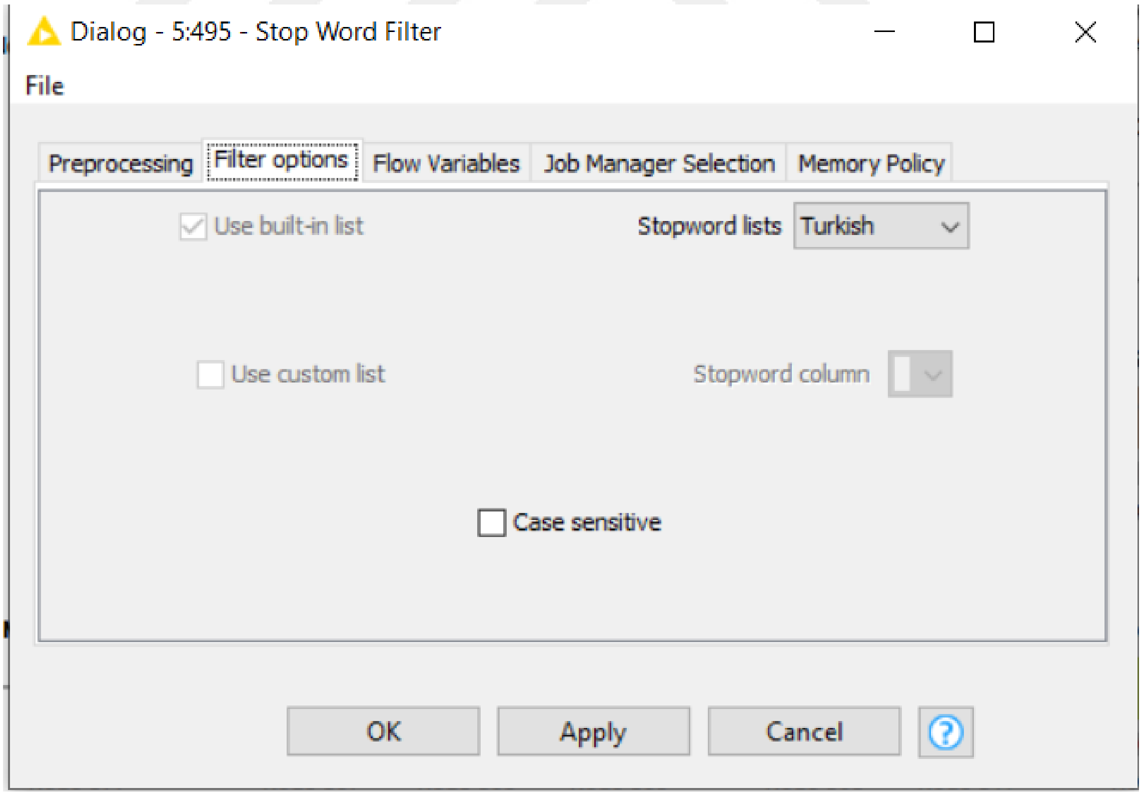
Figure A.17 KNIME Number Filter Node



Case to convert to

● To lower case ○ To upper case

Figure A.18 KNIME Case Converter Node



☑ Use built-in list

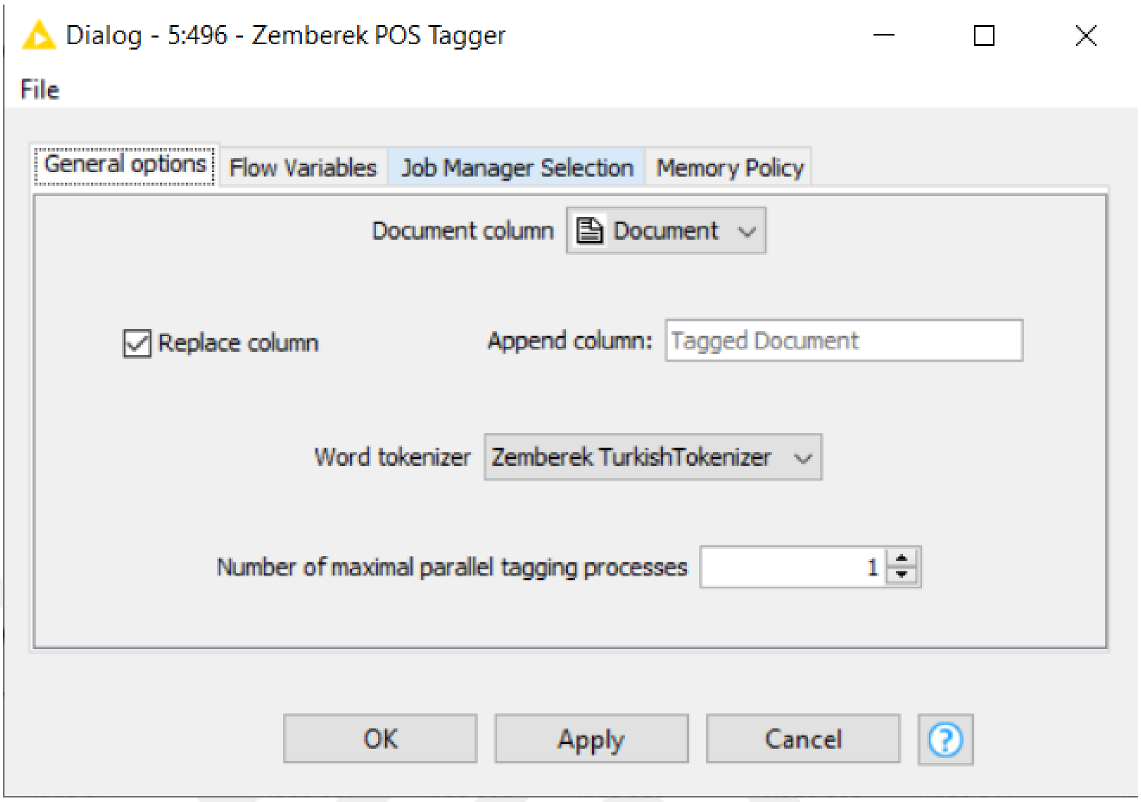
Stopword lists: Turkish

□ Use custom list

Stopword column

□ Case sensitive

Figure A.19 KNIME Stop Word Filter Node



Document column: Document

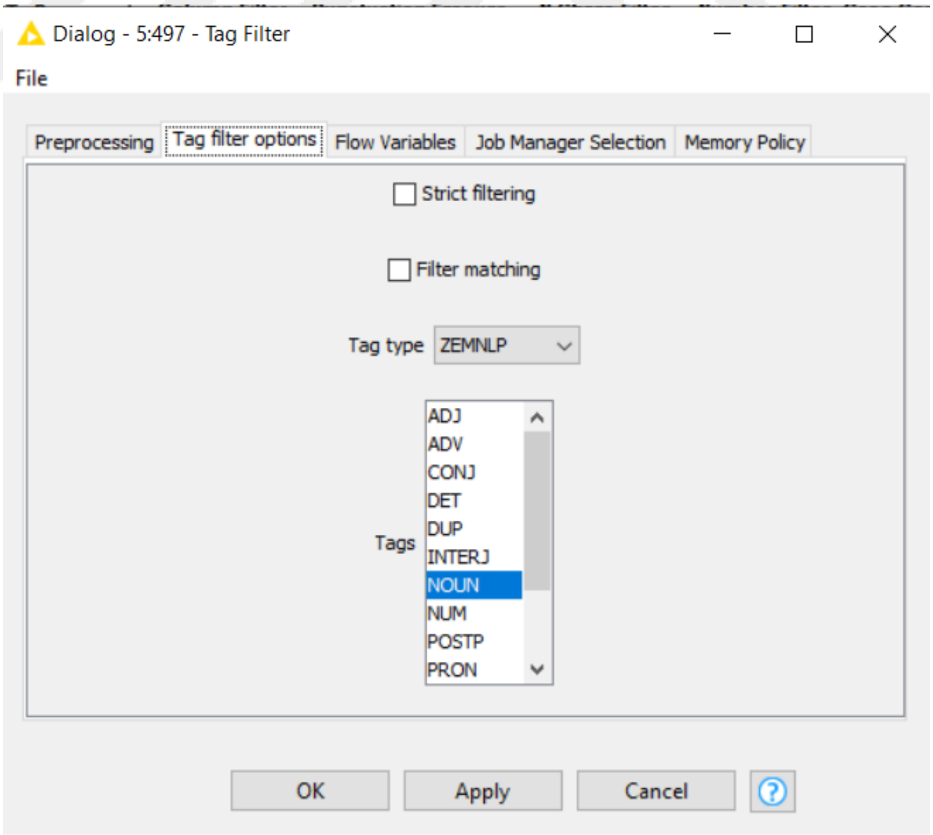
☑ Replace column

Append column: Tagged Document

Word tokenizer: Zemberek TurkishTokenizer

Number of maximal parallel tagging processes: 1

Figure A.20 KNIME Zemberek POS Tagger Node



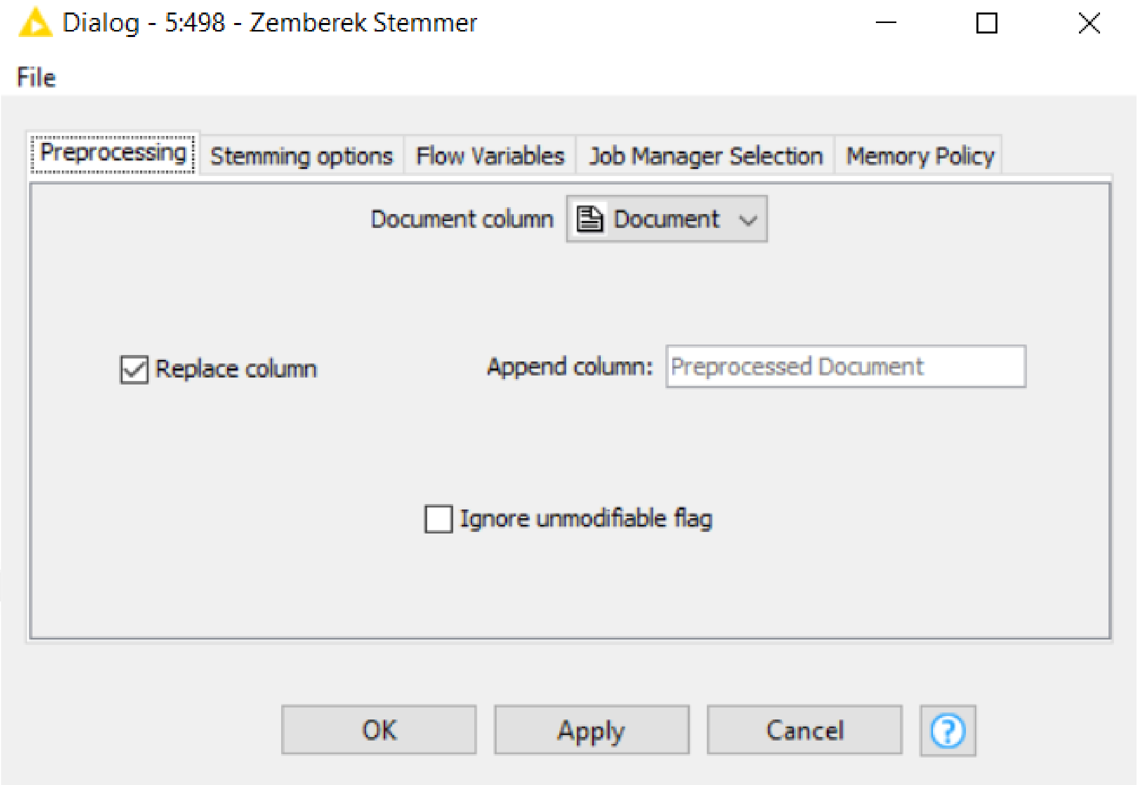
□ Strict filtering

□ Filter matching

Tag type: ZEMNLP

Tags: ADJ, ADV, CONJ, DET, DUP, INTERJ, NOUN, NUM, POSTP, PRON

Figure A.21 KNIME Tag Filter Node



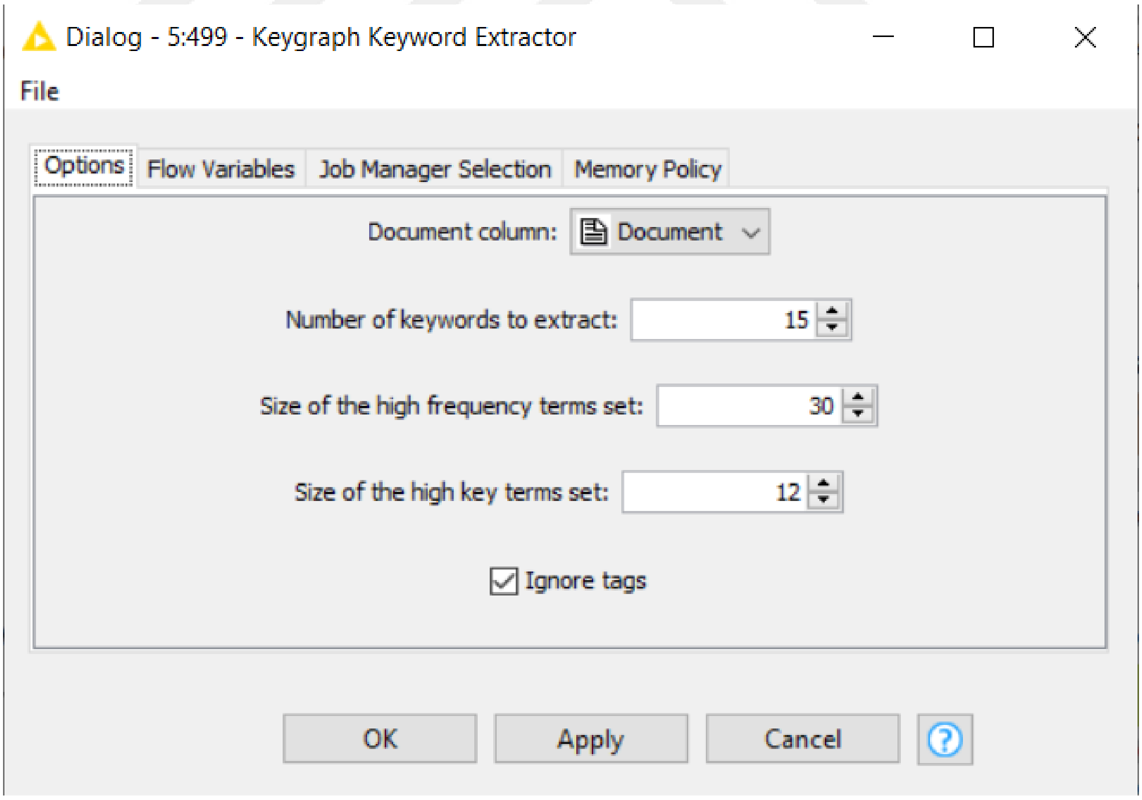
Document column: Document

☑ Replace column

Append column: Preprocessed Document

□ Ignore unmodifiable flag

Figure A.22 KNIME Zemberek Stemmer Node



Document column: Document

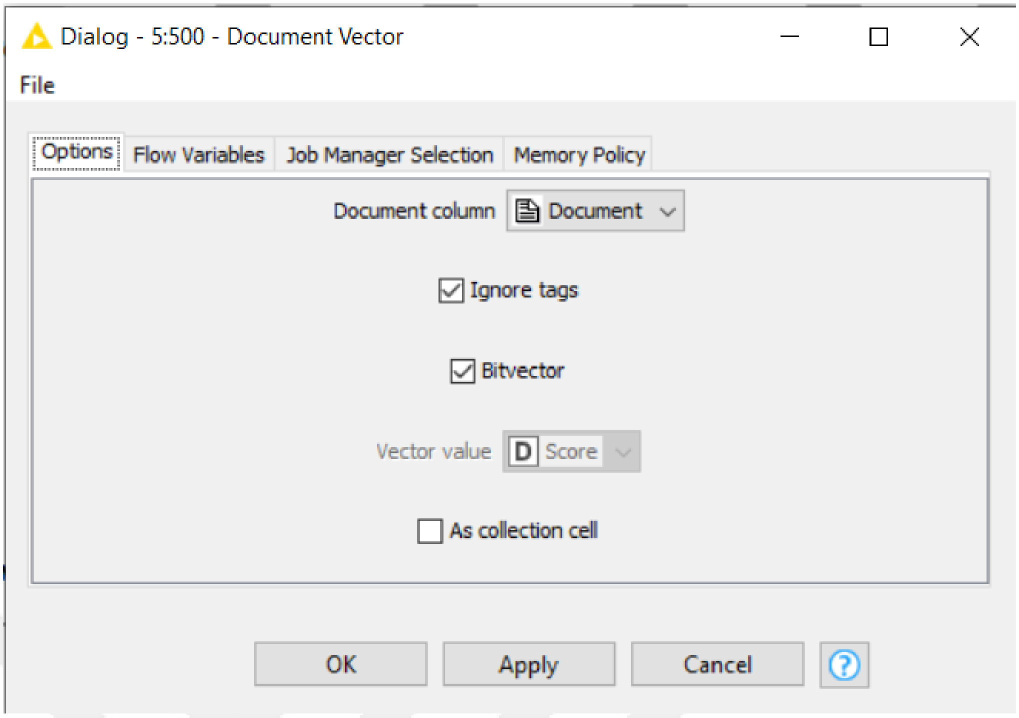
Number of keywords to extract: 15

Size of the high frequency terms set: 30

Size of the high key terms set: 12

☑ Ignore tags

Figure A.23 KNIME Keygraph Keyword Extractor Node



Dialog - 5:500 - Document Vector

Document column: Document

☑ Ignore tags

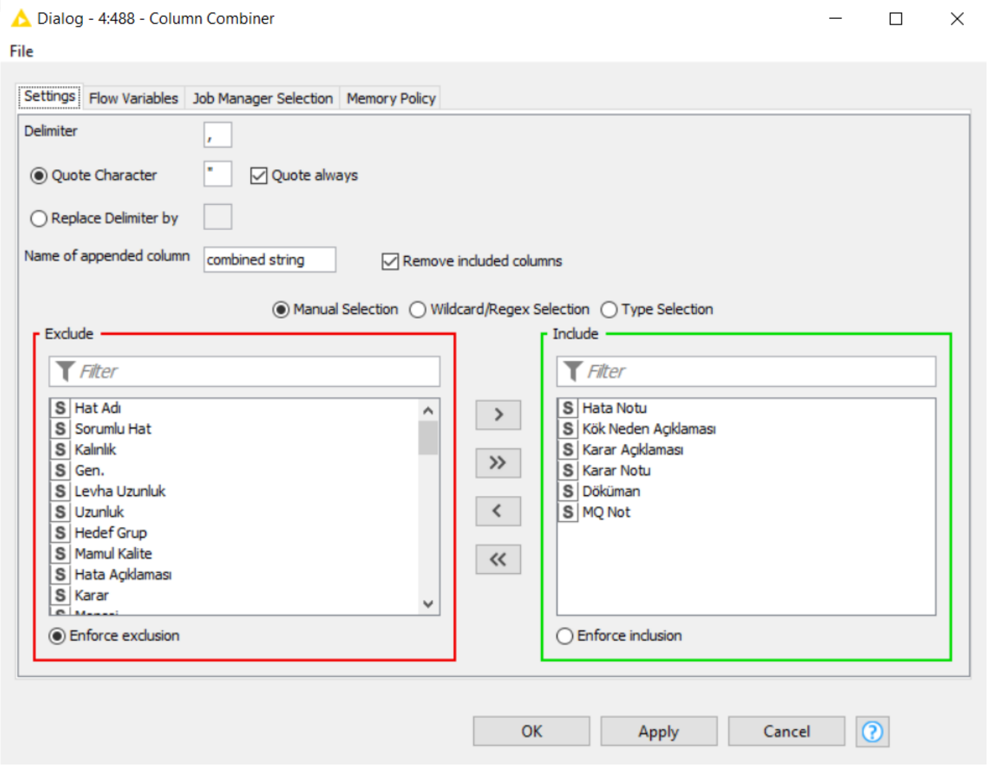
☑ Bitvector

Vector value: Score

□ As collection cell

Figure A.24 KNIME Document Vector Node

A.4 Combined Machine Learning Model



Dialog - 4:488 - Column Combiner

Delimiter: ,

● Quote Character: " ☑ Quote always

○ Replace Delimiter by

Name of appended column: combined string ☑ Remove included columns

● Manual Selection ○ Wildcard/Regex Selection ○ Type Selection

Exclude:
- S Hat Adı
- S Sorumlu Hat
- S Kalınlık
- S Gen.
- S Levha Uzunluk
- S Uzunluk
- S Hedef Grup
- S Mamul Kalite
- S Hata Açıklaması
- S Karar

● Enforce exclusion

Include:
- S Hata Notu
- S Kök Neden Açıklaması
- S Karar Açıklaması
- S Karar Notu
- S Döküman
- S MQ Not

○ Enforce inclusion

Figure A.25 KNIME Column Combiner Node

CURRICULUM VITAE

Personal Information
Name and surname: Asena ÖZTÜRK

Academic Background
Bachelor’s Degree Education: Industrial Engineering – METU - 2010
Foreign Languages: English

Work Experience
Institutions Served and Their Dates: Borçelik (2010-)