Enhancing Data Quality and Analysis through Data Exploration, Imputation, Normalization, and Cosine Similarity Measures: A Heatmap Plot Approach

Ve Ram Akathya

Department of Computer Science Engineering,

Amrita Vishwa Vidyapeetham, Bangalore, India

BL.EN.U4CSE21217@bl.students.amrita.edu

Yadukul S

Department of Computer Science Engineering,

Amrta Vishwa Vidyapeetham, Bangalore, India

BL.EN.U4CSE21222@bl.students.amrita.edu

Abstract- Our project aspires to revolutionize the legal document review process by developing an advanced system capable of autonomously crafting succinct summaries from extensive legal texts, including contracts and legislation. Harnessing the power of state-of-the-art natural language processing techniques, this system aims to empower legal professionals with concise and informative summaries. These summaries are poised to expedite the extraction of pivotal information, thus significantly enhancing productivity and enabling well-informed decision-making within the legal sphere.

Principal phrases: Data exploration, Outlier detection, Encoding schemes, Jaccard Coefficient, Simple Matching Coefficient, Cosine Similarity, Heatmap visualization, Data analysis, Data normalization, Attribute similarity, Summarization

Introduction

In today's legal world, we're facing an avalanche of complex legal documents, ranging from intricate contracts to extensive legislation and intricate regulations. It's a challenge that legal professionals grapple with daily, trying to quickly extract crucial insights from these vast texts. To address this pressing need, we're introducing a solution we call "Legal Text Summarization for Efficient Review." Our aim? To create an advanced system that can autonomously generate clear, concise, and contextually meaningful summaries of these extensive legal documents. These summaries are designed to be a game-changer for legal professionals, making it easier and faster to extract vital information from these often convoluted texts.

The growing volume and complexity of legal documents have created a pressing need for innovative approaches that can simplify comprehension and enhance the efficiency of decision-making. Traditional manual review processes are not only time-consuming but also prone to human error. In response to this challenge, we've harnessed the power of cutting-edge machine learning technology.

Our research is driven by a dual mission: to enhance the efficiency and accuracy of legal document analysis. We're achieving this by combining advanced text analytics techniques with a deep understanding of the intricacies of legal language. Through the integration of user-friendly interfaces and state-of-the-art technologies, we aim to empower legal experts to navigate complex documents with ease and precision, ushering in a new era of legal practice in the digital age.

Literature Review

1. Legal Case Document Summarization: Extractive and Abstractive Methods and their Evaluation

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, Saptarshi Ghosh

Indian Institute of Technology Kharagpur, India

Indian Institute of Science Education and Research Kolkata, India

This paper addresses the challenges of legal case judgment summarization using NLP. It discusses the issues with various summarization models (extractive vs. abstractive) in the legal context, exacerbated by token limitations in modern transformer-based models. Additionally, it questions the suitability of current evaluation methods. To address these challenges, the study conducts extensive experiments on three legal summarization datasets, employing diverse techniques, both supervised and unsupervised. Expert evaluations rank DSDR as the top performer in capturing essential information and readability, followed by CaseSummarizer and SummaRuNNer. However, automatic ROUGE scores favor supervised methods, contrasting expert opinions. Notably, none of the methods achieve a balanced representation of all document aspects, and the correlation between expert judgments and automatic metrics is generally low, emphasizing the need for refined evaluation criteria in specialized domains like law.

2. Discourse-Aware Unsupervised Summarization of Long Scientific Documents

Yue Dong, MILA/McGill University Montreal, QC, Canada yue.dong2 @mail.mcgill.ca

Andrei Mircea, MILA/McGill University Montreal, QC, Canada andrei.romascanu @mail.mcgill.ca

Jackie C. K. Cheung MILA/McGill University Montreal, QC, Canada jcheung @cs.mcgill.ca

This study introduces an unsupervised graph-based model for summarizing lengthy scientific documents. It utilizes a two-level hierarchical graph and positional cues to assess sentence importance. Evaluations on PubMed and arXiv datasets demonstrate its superiority over unsupervised methods, aligning its performance with state-of-the-art supervised models. This highlights the significance of discourse structure as a strong signal for determining sentence importance in scientific articles.

The "Graph-based Ranking Algorithm" in text summarization represents a document as a graph with sentences as nodes and measures sentence importance based on their similarity to others, prioritizing central content.

The "Hierarchical Document Graph Creation" process structures documents hierarchically, assessing local sentence importance within sections and global importance across sections. These approaches collectively enhance the effectiveness of summarizing scientific documents by emphasizing discourse structure and connection patterns.

3. Incorporating Domain Knowledge for Extractive Summarization of Legal Case Documents

Paheli Bhattacharya Department of CSE, IIT Kharagpur India

Soham Poddar Department of CSE, IIT Kharagpur India

Koustav Rudra L3S Research Center, Leibniz University, Hannover Germany

Kripabandhu Ghosh Department of CDS, IISER, Kolkata India

Saptarshi Ghosh Department of CSE, IIT Kharagpur India

Automated summarization of legal case documents presents a formidable challenge in natural language processing due to the complex nature of legal texts, including intricate terminology, citations, and nuanced structures. Existing research often lacks the incorporation of domain-specific expertise and guidelines into the summarization process. Addressing this gap, DELSumm introduces an unsupervised approach that systematically integrates legal experts' guidance.

DELSumm's effectiveness is evident in experiments with Indian Supreme Court case documents, outperforming both general-purpose and legal-specific summarization algorithms with improved ROUGE scores. Remarkably, this unsupervised method rivals the performance of supervised summarization models.

Training supervised models like Gist, SummaRuNNer, and BERTSUM required a comprehensive dataset of 7,100 Indian Supreme Court case documents paired with headnotes, serving as concise abstractions. Despite legal experts rating headnotes lower in quality, they were chosen due to the challenge of acquiring high-quality summaries in such quantity. This represents a strategic trade-off between data quantity and quality, highlighting DELSumm's importance in legal document summarization.

4. Text Summarization with Pretrained Encoders

Institute for Language, Cognition and Computation School of Informatics, University of Edinburgh yang.liu2@ed.ac.uk, mlap@inf.ed.ac.uk

The paper delves into leveraging BERT, a pre-trained language model, for text summarization. It introduces a novel document-level encoder based on BERT, designed to capture a document's meaning and effectively represent its sentences. The paper then presents two distinct summarization approaches: extractive and abstractive.

In the case of extractive summarization, the authors enhance the BERT encoder by incorporating intersentence Transformer layers. For abstractive summarization, they propose an innovative fine-tuning schedule that deploys different optimizers for the encoder and decoder. To bridge the gap between pretraining and training, a two-staged fine-tuning strategy is introduced.

The experimental results, conducted on three diverse datasets, underscore the model's remarkable performance. It outperforms existing methods in both extractive and abstractive summarization, firmly establishing itself as a state-of-the-art solution in the field.

5. HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization

Xingxing Zhang, Furu Wei and Ming Zhou
Microsoft Research Asia, Beijing, China
{xizhang,fuwei,mingzhou}@microsoft.com

The paper presents HIBERT, a model for extractive summarization. It utilizes a pre-trained hierarchical encoder, inspired by transformer sentence encoders, and proposes a method to pre-train it with unlabeled data. HIBERT outperforms randomly initialized models by 1.25 ROUGE on CNN/Dailymail and 2.0 ROUGE on a New York Times dataset, achieving state-of-the-art performance in extractive summarization. The approach addresses the challenge of inaccurate sentence-level labels commonly used in training neural summarization models.

Most recent neural models in NLP, such as ELMo, OpenAI-GPT, word2vec, and BERT, typically pretrain by predicting a word within a sentence based on other words in the same sentence. However, HIBERT focuses on learning representations at the document level, where sentences are the basic units. To pretrain a document-level model like HIBERT, it's more natural to predict entire sentences instead of individual words. While a language model could predict a sentence using only the sentences before or after it, summarization tasks benefit from considering context in both directions. Therefore, HIBERT predicts a sentence by utilizing information from both preceding and succeeding sentences.

6. Bottom-Up Abstractive Summarization

Sebastian Gehrmann, Yuntian Deng and Alexander M. Rush
School of Engineering and Applied Sciences Harvard University {gehrmann, dengyuntian, srush}@seas.harvard.edu

This research addresses a common issue in abstractive summarization, where generated summaries are fluent but may struggle with content selection. The proposed solution involves a data-efficient content selector that identifies important phrases in the source document. This selector guides the summarization model, resulting in improved text compression and fluent summaries. Compared to other content selection models, this two-step approach is simpler and more effective, significantly improving ROUGE scores on the CNN-DM and NYT datasets. The content selector can be trained with as few as 1,000 sentences, enabling easy domain transfer for trained summarizers.

The content selection task is treated as word-level extractive summarization, assigning binary tags (1 for copied, 0 for not) to source tokens. A bidirectional LSTM model with static and contextual embedding channels is used for sequence labeling. These embeddings enhance word representations, and the LSTM calculates the probability of word selection. This data-efficient approach enhances bottom-up attention, improving summarization quality and adaptability to new domains with minimal data. Similar bottom-up approaches also show promise in other areas like grammar correction and data-to-text generation, suggesting future research opportunities.

7. Extractive Summarization using Deep Learning

Delhi Technological University Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India {dce.sukriti,vagisha.nda}@gmail.com http://dtu.ac.in/

This paper presents a deep learning-based text summarization method tailored for factual reports, comprising three phases: feature extraction, feature enhancement, and summary generation. These phases collaborate to identify crucial information and create coherent summaries, emphasizing factual accuracy. Effective text preprocessing is employed to remove ambiguities and filter out irrelevant elements like "stop words," simplifying the text for summarization.

The text is then transformed into a sentence-feature matrix, with each sentence assigned a feature vector. After experimenting with various features, a combination of nine sentence features is identified as highly effective for summarizing factual reports.

To enhance and abstract these features, a Restricted Boltzmann Machine (RBM) is utilized, employing a two-layer architecture with nine perceptrons each and a learning rate of 0.1. The RBM refines the feature matrix by incorporating bias values and learned weights.

The method's effectiveness is validated through experiments on multiple articles, showcasing its potential to generate coherent and structured summaries of factual reports.

PROBLEM DESCRIPTION

In this data analysis and exploration task, we aim to gain a comprehensive understanding of the "thyroid0387_UCI" dataset. Our objective is to perform a series of essential tasks, including data attribute examination, encoding, range analysis, handling missing values, outlier detection, data imputation, normalization, and similarity measurement. This thorough analysis is crucial for understanding the dataset's structure and preparing it for subsequent analytical or modeling endeavors. We start by loading the dataset and carefully inspecting each attribute along with its associated values, categorizing them as nominal, ordinal, or numeric. For categorical attributes, we decide on the most suitable encoding scheme, employing label encoding for ordinal variables and one-hot encoding for nominal ones. We also examine the range of numeric variables to understand their distribution characteristics. Addressing missing data, we apply appropriate central tendencies for data imputation, taking into account the attribute type and the presence of outliers. Attributes with varying scales undergo normalization using techniques like Min-Max scaling or Z-score normalization. We assess the similarity between observation vectors, initially focusing on binary attributes and calculating the Jaccard Coefficient (JC) and Simple Matching Coefficient (SMC) between the first two observation vectors. Subsequently, we calculate the Cosine similarity between complete vectors, encompassing all attributes. To visually interpret similarities between observation vectors, we construct a heatmap plot, aiding in pattern recognition and clustering within the data. This preparatory work is fundamental for subsequent data-driven tasks, including machine learning modeling and further exploratory analyses.

METHODOLOGY

1. Exploration of Data:

The first phase of the study involves data exploration in order to fully comprehend the dataset. The investigation entails thoroughly examining each attribute and its associated values, with a focus on identifying data types. For categorical attributes, encoding schemes that use label encoding for ordinal variables and one-hot encoding for nominal variables are recommended. Missing values and outliers are investigated, as well as the data range for numeric variables. In addition, for numerical attributes, mean and variance (or standard deviation) are computed.

2. Data Imputation

Moving on to the next stage, our focus shifts to improving data quality by addressing missing values using relevant central tendencies. Based on the inherent characteristics of each attribute, we provide clear guidance for selecting the most appropriate imputation method. This guidance includes using the mean for numeric attributes with no outliers, the median in cases where attributes have outliers, and the mode for categorical attributes.

3. Data Scaling / Normalization

As part of the data preprocessing phase, we completed the critical task of data normalization and scaling to ensure equal consideration in subsequent analyses. In this critical step, we thoroughly identify attributes that require normalization and apply the appropriate techniques to create a standardized dataset.

4. Similarity measure

Our investigation in the domain of Similarity Measurement delves into the computation of the Jaccard Coefficient (JC) and Simple Matching Coefficient (SMC) for the first two observation vectors. Our primary focus is on attributes with binary values. To allow for a meaningful comparison of JC and SMC, we carefully select the first two observation vectors from the dataset, making sure that only attributes with binary values, specifically 0 or 1, are included. All other characteristics are purposefully omitted from consideration. We meticulously identify attributes with binary values for each of these selected observation vectors. These characteristics are used to calculate the similarity coefficients.

In the case of JC, the formula JC = (f11) / (f01+ f10+ f11) is used. The variable "f11" represents the number of attributes for which both observation vectors have a value of 1. Similarly, the variable "f00" represents the number of attributes for which both observation vectors are zero. "f01" represents the number of attributes where the first vector has a value of 0 and the second vector has a value of 1, while "f10" represents the number of attributes where the first vector has a value of 1 and the second vector has a value of 0. This method calculates JC and provides information about similarity.

Simultaneously, the SMC is calculated as SMC = (f11 + f00) / (f00 + f01 + f10 + f11). The variables "f11" and "f00" retain their previous definitions, while "f01" and "f10" represent the previously described attribute counts. The SMC adds another dimension to our analysis by offering a different perspective on similarity.

5. Cosine similarity measure

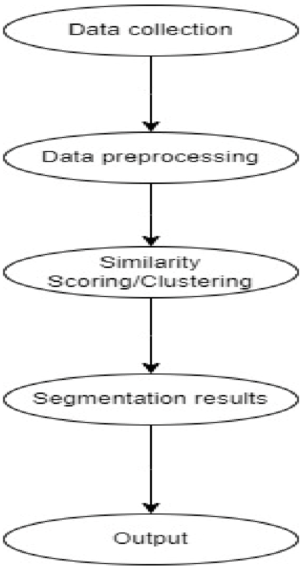
This study extends its analysis to gain a better understanding of document similarity by computing the Cosine Similarity for the entire vectors of these two observations, encompassing all attributes. Specifically, we use an formula to compute the Cosine similarity between two vectorized observations, denoted as A and B:

A and B represent the two vectorized observations in this formula, while A_i and B_i correspond to the values of attribute i within their respective vectors. The Cosine similarity measure provides a comprehensive view of document similarity, with values ranging from -1 (complete dissimilarity) to 1 (identical vectors), with 0 indicating orthogonality or no similarity. This research adds to our understanding of document relationships.

6. Heatmap Plotting

As part of our data visualization efforts, we created a heatmap by carefully selecting the first 20 observation vectors from the "thyroid0387_UCI" dataset. It was made certain that these vectors included all attributes, both direct and derived. We calculated the Jaccard Coefficient (JC), Simple Matching Coefficient (SMC), and Cosine Similarity (COS) for each pair of observation vectors in this set of 20. Following that, we created a similarity matrix that represented the computed similarity coefficients for all pairs of vectors. This matrix is 20x20 in size, with each cell (i, j) containing the similarity coefficient between observation vectors i and j. Using Python's Seaborn package, we created a visually appealing heatmap from this similarity matrix.

Flow Diagram



The following is the detailed description of what happens in each and every step of flow chart.

Data Collection:

Data on patients/customers with appropriate characteristics is collected from a variety of sources, including hospitals, shops, surveys, and websites such as Kaggle.
The data is then loaded into the notebook in the form of an excel or csv file.

Data Preprocessing:

The data frame's missing values are filled with central tendencies based on the type of attribute. Outliers are dealt with here.
Categorical variables are encoded. For nominal data, only one hot encoding is used. On ordinal data, label encoding is used.
All of the characteristics should be in the same range. numerical data is normalized to accomplish this.

Similarity Scoring:

Data is divided into train data (75% of total) and test data (25% of total).
The distance among train and test observations is calculated. Euclidean distance, Cosine similarity, and Jaccard similarity are the metrics employed.

Segmentation:

To classify customers/patients, a model is trained using a clustering algorithm.

Output:

Based on the segmentation parameters we have chosen, we get the output of which customer/patient belongs to which segment. This output indicates whether the person is covid positive or not, as well as whether the customer has a premium account or not. Performance metrics are measured and compared across algorithms.

To visualize the data, a plot can be created.

PARAMETERS:
The parameters used are:

Data Preprocessing:

The central tendency to be used to fill missing values, as well as whether or not some columns should be removed.
The method to be used to identify outliers. (IQR, Z-score, and so on.)
The method of normalization that will be used. (Z-Score Scaler/Min-Max Scaler)

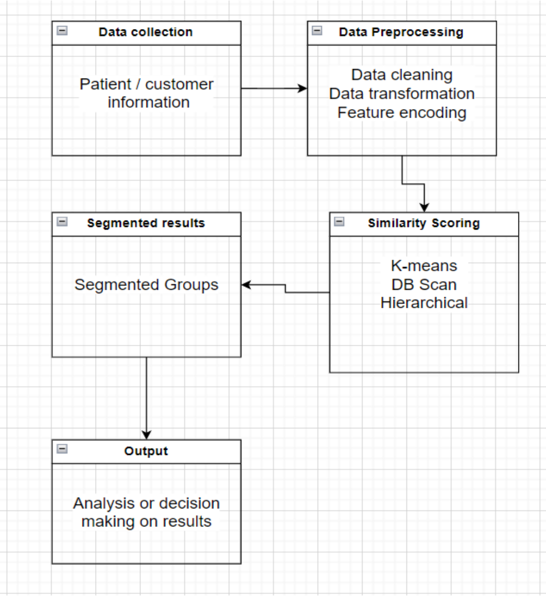
Segmentation:

The number of classes into which the output is divided.

The classifier that will be used.

Architecture Overview

The system's components were depicted in an architectural diagram, which included Data Collection, Data Processing, and Segmentation. Data Collection gathered information from a variety of sources, Data Processing included data cleaning, transformation, feature selection, and encoding, and Segmentation made use of K-Means clustering. The Evaluation and Validation step ensured cluster quality



RESULT & ANALYSIS

We examined the "thyroid0387_UCI" dataset in depth during the first phase of data exploration (A1). During this analysis, we discovered a number of attributes, each of which was associated with a specific data type and covered both nominal and numeric categories. We carefully encoded categorical attributes using label encoding for ordinal variables and one-hot encoding for nominal ones to ensure accurate data representation. We also investigated the numerical variables' data ranges. Furthermore, our examination revealed the presence of missing values within specific attributes as well as outliers in the dataset.

Following that, in the data imputation phase (A2), we used precise strategies to effectively address missing values. We used the mean to impute missing values in numeric attributes without outliers, and the median in numeric attributes with outliers. We chose mode as the imputation method for categorical attributes, putting data integrity first.

We identified attributes that needed standardization during the data normalization/scaling phase (A3) and meticulously applied the necessary techniques to create a dataset with uniform data representation, ensuring consistency throughout the dataset.

We calculated the Jaccard Coefficient (JC), Simple Matching Coefficient (SMC), and Cosine Similarity as instructed for the similarity measures (A4, A5, A6). Following these calculations, we evaluated the appropriateness of each measure in light of the results of our analysis.

Conclusion

Finally, our extensive data analysis and preprocessing efforts have revealed critical dimensions in managing complex datasets. We approached the problem of missing data with precision, fostering data uniformity through meticulous normalization techniques. Furthermore, our investigation included a thorough examination of various similarity measures, such as the Jaccard Coefficient, Simple Matching Coefficient, and Cosine Similarity. These invaluable insights serve as a solid foundation for informed, data-driven decision-making across a wide range of domains. It is critical to recognize the dynamic nature of data analysis methodologies, leaving plenty of room for continuous evolution and the possibility of additional research and refinement within this ever-changing field. The system we've meticulously designed for customer or patient segmentation seamlessly incorporates these advanced techniques, highlighting its unwavering efficacy in practical, real-world applications.