Methodology for Analyzing Educational Forums with NLP: Searching for Economic Terms

José Javier Galán Hernández[1], Gabriel Marín Diaz[2], Gonzalo Mariscal[3]

1 Professor, Department of Information Systems and Computer Science Faculty of Statistical Studies, Universidad Complutense, Avenida Puerta de Hierro, s/n, 28040 Madrid. Correo-e: josejgal@ucm.es

2 Professor, Department of Information Systems and Computer Science Faculty of Statistical Studies, Universidad Complutense, Avenida Puerta de Hierro, s/n, 28040 Madrid. Correo-e: gmarin03@ucm.es

3 Head of Department, Department of Computing and Technology, School of Architecture, Engineering and Design, Universidad Europea de Madrid, C/Tajo, S/N. 28670-Villaviciosa de Odon (Madrid). Correo: Gonzalo.mariscal@universidadeuropea.es

Abstract

This document studies the programming languages and libraries suitable for presenting a methodology for analyzing forums in the economics subject using natural language processing (NLP) techniques, concluding to use spaCy and transformers in Python. The methodology follows a structure based on CRISP-DM, including project planning and the selection of appropriate tools and technologies. The proposed methodology performs the following actions:

Relevant data sources are identified and accessed, collecting data from forum posts, such as text, dates, and authors. Text preprocessing involves noise removal, tokenization, and lemmatization using spaCy, ensuring clean and manageable data.

Content analysis begins with calculating the frequency of key terms, followed by topic modeling with techniques like LDA to identify the main discussion topics. Sentiment analysis is performed with transformers models to evaluate the tone of the posts.

The results are communicated through visualizations such as word clouds and bar charts, providing a clear understanding of the data. The results are documented in detailed reports that describe the methods used and the interpretations of the findings.

Lastly, the results are analyzed and discussed in relation to the initial objectives of the project, offering conclusions and recommendations for future actions or additional studies.

1. Introduction

The analysis of textual data has become an essential tool for extracting valuable information from large volumes of text generated on digital platforms [1]. In the educational context, discussion forums represent a rich data source that can provide significant insights into student interactions and learning [2]. This document proposes a methodology tailored for analyzing economics subject forums using NLP techniques with the spaCy and transformers libraries in Python.

The proposed methodology follows a structure based on the CRISP-DM (Cross Industry Standard Process for Data Mining) standard[3], adapted for text analysis. The steps include project planning, tool and technology selection, and a systematic content analysis process. Initially, relevant data sources are identified and accessed, extracting crucial information from forum posts, such as text, dates, and authors.

Data preprocessing ensures noise removal and prepares the text for analysis using tokenization and lemmatization techniques provided by spaCy. Next, content analysis is performed, including calculating the frequency of key terms and topic modeling using LDA, allowing the identification of the main discussion topics. Additionally, sentiment analysis is conducted with transformers models to evaluate the tone of the posts.

The analysis results can be visualized through word clouds and bar charts and are documented in detailed reports that describe the methods and findings. Finally, the results are discussed in the context of the project objectives, offering conclusions and recommendations. This methodology provides a foundation for gaining insights from discussions in economics forums, contributing to a better understanding of student behavior and concerns.

2. State of the Art

There are various scientific publications related to "NLP" and "methodologies" for adapting it to a specific discipline. In May 2024, the Scopus database [4], a database widely recognized by the scientific community, was consulted, yielding 21 results, see Table 1 [5-25], but none applied to methodologies for analyzing educational forums on economics.

Table 1. Table of scientific documents related to “methodology” and “NLP”

Title	Authors	Year
Innovative agricultural ontology construction using NLP methodologies and graph neural network	Sanju Saravanan K.; Bhagavathiappan V.	2024
Systems Engineering Process Enhancement: Requirements Verification Methodology using Natural Language Processing (NLP) for Automotive Industry	Júnior F.S.; Reis P.A.; Cavalcante M.S.; De Oliveira A.H.M.	2024
Retrieval methodology for similar NPP LCO cases based on domain specific NLP	Seong N.K.; Lee J.H.; Lee J.B.; Seong P.H.	2023
An NLP-based statistical reporting methodology applied to court decisions	Bellandi V.; Maghool S.; Siccardi S.	2023
A critical assessment of consumer reviews: A hybrid NLP-based methodology	Biswas B.; Sengupta P.; Kumar A.; Delen D.; Gupta S.	2022

A Methodology for Enabling NLP Capabilities on Edge and Low-Resource Devices	Goulas A.; Malamas N.; Symeonidis A.L.	2022
A NLP-Oriented Methodology to Enhance Event Log Quality	Ramos-Gutiérrez B.; Varela-Vaca Á.J.; Ortega F.J.; Gómez-López M.T.; Wynn M.T.	2021
Explainable NLP: A Novel Methodology to Generate Human-Interpretable Explanation for Semantic Text Similarity	De T.; Mukherjee D.	2021
Basic Methodologies Used in NLP Area	Wang Y.	2020
A two-stage LP-NLP methodology for the least-cost design and operation of water distribution systems	Qiu M.; Housh M.; Ostfeld A.	2020
Internet data analysis methodology for cyberterrorism vocabulary detection, combining techniques of big data analytics, NLP and semantic web	Castillo-Zúñiga I.; Luna-Rosas F.J.; Rodríguez-Martínez L.C.; Muñoz-Arteaga J.; López-Veyna J.I.; Rodríguez-Díaz M.A.	2020
Analyse digital forensic evidences through a semantic-based methodology and NLP techniques	Amato F.; Cozzolino G.; Moscato V.; Moscato F.	2019
Design and construction of a NLP based knowledge extraction methodology in the medical domain applied to clinical information	Moreno D.C.; Vargas-Lombardo M.	2018
An instrumented methodology to analyze and categorize information flows on twitter using nlp and deep learning: A use case on air quality	Juanals B.; Minel J.L.	2018
Nlp methodology as guidance and verification of the data mining of survey ensanut 2012	Vargas V.M.C.; Stephens C.R.; Martínez G.E.S.; Rendón A.M.	2015
Rule-based NLP methodology for semantic interpretation of impact factors for construction claim cases	Niu J.; Issa R.R.A.	2014
An integrated AHP-NLP methodology for facility layout design	Hadi-Vencheh A.; Mohamadghasemi A.	2013
A comparative survey on NLP/U methodologies for processing multi-documents	Mills M.T.; Bourbakis N.G.	2012

Mining methodologies from NLP publications: A case study in automatic terminology recognition	Kovačević A.; Konjović Z.; Milosavljević B.; Nenadic G.	2012
Methodology to develop and evaluate a semantic representation for NLP.	Irwin J.Y.; Harkema H.; Christensen L.M.; Schleyer T.; Haug P.J.; Chapman W.W.	2009
Medical i2b2 NLP Smoking Challenge: The A-Life System Architecture and Methodology	Heinze D.T.; Morsch M.L.; Potter B.C.; Sheffer Jr. R.E.	2008

Table 1 presents a variety of studies that have developed methodologies using natural language processing (NLP) in different contexts and applications. However, none of these studies specifically focus on applying NLP to the analysis of educational forums in economics. This gap in the literature suggests the need for new research that addresses this specific area. The table's content is related to the need to develop an appropriate methodology for this purpose as follows:

Diversity of Applications:

The presented studies cover a wide range of NLP applications, from agricultural ontology construction [5] to the analysis of judicial decisions [8]. However, no methodology applied to the analysis of educational forums, specifically in the field of economics, is identified.

Methodologies in Specific Contexts:

Most studies focus on very specific contexts, such as the automotive industry [6], cybersecurity [26], and medicine [17]. These works show how NLP can be adapted to particular needs but do not provide a generalizable approach for the analysis of educational forums.

Lack of Focus on Education:

Although there are studies on improving the quality of event logs [11] and evaluating information flows on social networks [27], there is no clear methodology for applying NLP to the analysis of educational content in economics forums. This is critical since educational forums represent a rich source of qualitative data that can provide valuable insights into student interaction and learning.

Need for Integration of NLP Techniques:

The reviewed works use various NLP techniques, such as machine learning for semantic interpretation [12] and text processing on low-capacity devices [10]. These techniques could be useful for developing a specific methodology for forum analysis, combining elements like sentiment analysis, topic identification, and evaluation of student interaction.

The review of the existing literature shows that, while there are various NLP methodologies developed for specific contexts, none directly address the analysis of

educational forums in economics. This creates an opportunity to develop a specific methodology that combines the best NLP practices with techniques adapted to the qualitative and dynamic nature of educational forums.

Creating a specific methodology to apply NLP to the analysis of educational forums in economics is necessary and justified. The table shows that current methodologies do not cover this area, suggesting a gap in the literature that could be filled with dedicated research. This new methodology could leverage advanced NLP techniques to provide valuable insights into forum dynamics, improve the understanding of student learning, and support educators in enhancing their teaching practices.

The first step is to understand the tools and technologies, in relation to NLP, that best suit the needs of this project.

3. Selection of Tools and Technologies

Continuing from the introduction, an analysis of existing languages and technologies related to NLP is presented, along with an introduction to it.

3.1 Natural Language Processing (NLP)

NLP is a field of artificial intelligence that focuses on the interaction between computers and human language. The goal of NLP is to enable computers to understand, interpret, and respond to human language inputs in a valuable and useful manner. To integrate NLP, we need to know which programming languages support it and select the appropriate one [28].

3.2 Selection of the Programming Language

NLP can be performed in various programming languages, as shown in Table 1.

Table 1. Table of Programming Languages Used in NLP

Language	Description	Libraries	Comments
Python	Interpreted language, very popular in data science and NLP	NLTK, spaCy, TextBlob, Gensim, transformers	Extensive resources and community support
Java	Compiled language, widely used in enterprise applications	Apache OpenNLP, Stanford NLP	Good integration in enterprise systems, solid performance
R	Language and environment for statistical computing and graphics	tm, text, syuzhet	Ideal for statistical analysis and data visualization
JavaScript	Interpreted language, essential for web development	Natural	Used in web applications and server-side development

C#	Object-oriented programming language used in Windows applications	Stanford NLP (through ports)	Good integration in the Microsoft ecosystem and Windows applications

Python is the chosen language for this NLP project for several key reasons, all of which are reflected in Table 1:

Popularity and Support Community:

Python is one of the most popular languages in data science and NLP. This translates to a vast amount of available resources, from detailed documentation to discussion forums and online tutorials. The active community means it is easier to find solutions to common problems and get help when needed.

Extensive and Powerful Libraries:

Python has a wide variety of specialized libraries for natural language processing, such as NLTK, spaCy, TextBlob, Gensim, and transformers. These libraries cover all NLP needs, from text preprocessing to advanced modeling and sentiment analysis, providing comprehensive and efficient tools for every phase of the project.

Flexibility and Ease of Use:

Python is an interpreted, high-level language, which makes code writing and reading easier. Its simple and clear syntax allows developers to focus on solving NLP problems without worrying about the complexity of the language. This is especially useful in educational and research projects where clarity and rapid development are crucial.

Integration with Other Data Science Ecosystems:

Python is not only powerful for NLP but also integrates perfectly with other data science and machine learning tools and libraries, such as Pandas, NumPy, and scikit-learn. This integration allows for more comprehensive and sophisticated analyses, combining NLP techniques with predictive models and statistical analyses.

Support for Deep Learning:

Python libraries like transformers and TensorFlow/Keras enable the implementation of deep learning models, which are essential for advanced NLP tasks such as machine translation, text generation, and large-scale sentiment analysis. This support makes Python a robust choice for projects requiring cutting-edge models.

Portability and Compatibility:

Python is cross-platform and can run on various operating systems, including Windows, macOS, and Linux. This facilitates collaboration in multidisciplinary teams and the deployment of solutions in different environments without significant adjustments.

Therefore, Python is the ideal choice for the NLP project focused on analyzing educational forums in economics due to its popularity, extensive range of libraries, ease of use, integration with other data science tools, and support for advanced deep learning techniques. These characteristics ensure that the project can be carried out efficiently and effectively, maximizing resources and obtaining high-quality results. [27-29].

3.3 Selection of Natural Language Processing (NLP) Technologies with Python

Focusing on the use of the Python language, we analyze the most popular NLP libraries (see Table 2), deciding that for analyzing educational forums on economics, spaCy and transformers (Hugging Face) are the best options due to their advanced capabilities, efficiency, and accuracy [30-33].

Table 2. Table of NLP Tools for Python

Tool	Description	Main Functions	Comments
NLTK (Natural Language Toolkit)	Comprehensive library for NLP	Tokenization, Lemmatization, POS Tagging, NER, Sentiment Analysis	Extensive range of functions, good for learning and prototyping
spaCy	Advanced and high-performance NLP library	Tokenization, POS Tagging, NER, Dependency Parsing	Very fast and efficient, good for production use
TextBlob	Simple library built on NLTK and Pattern	Sentiment Analysis, Translation, Spelling Correction	Easy to use, good for simple tasks
Gensim	Library for topic modeling and document similarity	Topic Modeling (LDA), Word Embeddings	Ideal for topic modeling and similarity analysis
transformers (Hugging Face)	Library for state-of-the-art NLP models	Text Classification, Sentiment Analysis, Text Generation	Uses pre-trained models like BERT, GPT-3; very powerful

The combination of spaCy and transformers for the natural language processing (NLP) project focused on analyzing academic forums on economics is highly recommended for several key reasons:

Complementarity of Functionalities:

spaCy: SpaCy is a robust and efficient library for text preprocessing. It offers advanced tools for tokenization, lemmatization, part-of-speech tagging (POS tagging), and dependency parsing. These functionalities are essential for preparing textual data before applying more complex techniques.

Transformers: Developed by Hugging Face, this library provides access to state-of-the-art NLP models such as BERT, GPT-3, and RoBERTa. These models are ideal for advanced tasks like sentiment analysis, text classification, and text generation.

Ease of Integration:

SpaCy and transformers are designed to work well together. SpaCy can be used for efficient and structured data preprocessing, while transformers can apply deep learning models for specific tasks. This integration allows for a smooth and optimized workflow.

SpaCy allows for the direct integration of transformer models into its pipeline, facilitating the use of advanced models without the need for complex intermediate steps. This simplifies the development process and improves efficiency.

Performance and Efficiency:

SpaCy is known for its speed and efficiency in handling large volumes of textual data. It is especially suitable for real-time applications where speed is crucial.

Although transformer models can be computationally intensive, they offer high-precision results and are capable of capturing complex nuances in language, which is essential for deep and accurate text analysis.

Extensive Community and Resources:

SpaCy has an active community and comprehensive documentation, which makes problem-solving and continuous learning easier. Additionally, spaCy is regularly updated with new features and improvements.

Hugging Face offers a collaborative platform with pre-trained models, tutorials, and deployment tools. The Hugging Face community is very active, providing continuous support and resources.

Practical Applications in NLP:

Using models like BERT through transformers, precise sentiment analysis can be performed to understand the emotions and opinions of students in forums.

With spaCy for preprocessing and transformers for advanced modeling, efficient and accurate topic identification can be achieved, providing insights into the main areas of interest and discussion in forums.

The combination of spaCy and transformers is highly recommended for the NLP project focused on analyzing academic forums on economics due to their complementarity, ease of integration, efficient performance, and extensive community support and resources. These tools allow for the implementation of a robust and optimized workflow, ensuring precise and useful results for the analysis of textual data.

Now that we have the appropriate programming technology for NLP, we can propose a methodology that uses it.

3.4 Adaptation of a Data Mining Methodology

To propose a methodology, we need to use an existing methodology that can be adapted to our needs. The CRISP-DM model [34] is a widely used methodology for conducting data mining projects. It is structured into six key phases, providing a comprehensive and systematic framework for planning, executing, and evaluating data analysis projects. These phases are iterative and flexible, allowing adjustments and refinements at any stage of the process. Below are the phases of the CRISP-DM model:

Business Understanding:

Understand the project's objectives and requirements from a business perspective. Determine business objectives, assess the situation, define data mining goals, and develop an initial project plan.

Data Understanding:

Collect and familiarize yourself with the available data to identify quality issues and gain initial insights. Collect initial data, describe the data, explore the data, and verify data quality.

Data Preparation:

Prepare the final data to be used in the modeling phase. Select relevant data, clean the data, construct new attributes, integrate data from multiple sources, and format the data appropriately.

Modeling:

Apply modeling techniques and calibrate parameters to obtain the best results. Select modeling techniques, design tests, build models, and evaluate models.

Evaluation:

Thoroughly evaluate the models to ensure they meet business objectives and are ready for deployment. Evaluate results, review the data mining process, and determine the next steps.

Deployment:

Implement the model results in a production environment where they can be used for decision-making. Plan deployment, monitor and maintain models, produce final reports, and review the project.

Now we have a methodology that can be adapted to the project's needs.

4. Methodology

Using the aforementioned technology with Python, and based on the CRISP-DM methodology while also incorporating elements of the scientific research methodology, the methodology proposed in Figure 1 is presented. Each of the processes is described below and will be supported by the Python data science language.

Figure 1. Proposed methodology

Finally, we can observe how each phase of the proposed methodology is related to a phase of the original CRISP-DM methodology

Table 3. Comparison table between CRISP-DM methodology and proposed methodology

Aspect	CRISP-DM	Proposed Methodology
Main Phases	6 phases	5 phases
Phase 1	Business Understanding	Data Collection
	Determine business objectives	Identify and access relevant data sources
	Assess the situation	Extract data from posts (text, dates, authors, etc.)
Phase 2	Data Understanding	Text Preprocessing
	Collect initial data	Clean the text (remove noise, HTML, links, special characters)
	Describe data	Tokenize and lemmatize the text
	Explore data	
	Verify data quality	
Phase 3	Data Preparation	Content Analysis
	Select data	Calculate the frequency of key terms
	Clean data	Apply topic modeling techniques (e.g., LDA)
	Construct new attributes	
	Integrate data	

	Format data	
Phase 4	Modeling	Report Generation
	Select modeling techniques	Create visualizations (word clouds, bar charts)
	Generate test design	Draft detailed reports
	Build models	
	Evaluate models	
Phase 5	Evaluation	Results and Discussion
	Evaluate results	Interpret and discuss results in relation to objectives
	Review process	Provide a deep understanding of the findings
	Determine next steps	Make recommendations for future actions or research
Phase 6	Deployment	(Integrated in previous phases)
	Plan deployment Plan monitoring and maintenance Produce final report Review project	

CRISP-DM is a methodology recognized for its flexibility and iterativity, allowing for the return to previous phases as necessary to refine and improve project results. Originally designed for general data mining, this model is adaptable enough to be applied to specific projects involving various types of data and objectives. A notable aspect of CRISP-DM is its focus on business understanding, emphasizing the importance of understanding business objectives and how data can contribute to achieving them. This approach ensures that data mining projects not only focus on technical aspects but also on generating value for the organization from the data.

On the other hand, the proposed methodology for text analysis specializes in handling textual data, adapting each phase to manage the particularities of text. This methodology simplifies some phases of CRISP-DM, combining modeling and evaluation

into the stages of report generation and discussion of results, allowing for a more direct and focused workflow. Additionally, it focuses on practical and specific steps for the preparation and analysis of texts, emphasizing the importance of text cleaning and preprocessing. The creation of useful reports for decision-making is a key component, ensuring that findings are presented in a clear and applicable manner. This practical and specialized approach facilitates the extraction of valuable insights from textual data, optimizing its relevance and utility in specific contexts.

Below, each phase of the proposed methodology is explained.

4.1 Data Collection

Identify and access relevant data sources, in this case, forums on the subject of economics. It is crucial to select forums that are representative and contain meaningful discussions on topics of interest.

Obtain data from the posts, including text, dates, authors, etc. This step involves the systematic collection of all relevant information from the forum posts, ensuring that all the necessary details are captured for subsequent analysis [35].

4.2 Text Preprocessing

It is important to perform data cleaning; removing noise such as HTML, links, special characters, etc. This step is essential to ensure that the data is clear and manageable, eliminating elements that do not add value to the analysis.

Use spaCy and Transformers to tokenize and lemmatize the text. Tokenization breaks down the text into smaller units (tokens), while lemmatization converts the tokens to their base form or lemma, facilitating subsequent linguistic analysis.

4.3 Content Analysis

Calculate the frequency of key terms to determine which words or phrases are most frequently used in the data corpus, providing an initial view of the predominant topics.

Use techniques such as LDA to identify main topics. Topic modeling helps discover groups of words that tend to appear together, revealing the underlying themes in forum discussions.

Perform sentiment analysis to determine if the tone of the text is positive, negative, or neutral.

4.4 Report Generation

Create visualizations such as word clouds and bar charts to communicate the findings. These visualizations allow for a quick and effective understanding of the data, highlighting the most frequent terms and the main topics identified in the analysis.

Write detailed reports that include the methods used, the results obtained, and the interpretations of these results. The documentation should provide a clear explanation

of the processes and findings, facilitating the understanding and use of the results for decision-making or future research.

4.5 Results and Discussion

Analyze and discuss the results obtained in the context of the initial objectives of the project. This step involves interpreting the data in relation to the original questions and goals, providing a deep understanding of the findings [36].

Offer conclusions based on the analysis and suggest possible actions or areas for future studies. Here, the key points of the analysis are summarized, the implications of the results are presented, and recommendations for improvements or additional research are proposed.

5. Base Code

This code provides a solid foundation for text analysis, serving as an educational example, covering everything from preprocessing and tokenization to sentiment analysis and topic classification. However, to fully meet the original goal of identifying specific texts from economics forums, the classification model needs to be adapted and trained with specific data from economic forums.

The provided code uses NLP techniques to perform topic analysis and sentiment analysis on texts. LDA identifies the predominant topics in the texts, while transformer models evaluate the tone of the text, providing a clear and understandable view of the content and emotion in the analyzed texts.

5.1 Code

This code simulates receiving text from a forum for analysis.

```
import spacy
from gensim import corpora
from gensim.models import LdaModel
from transformers import pipeline
import matplotlib.pyplot as plt
from wordcloud import WordCloud

# Sample texts
texts = [
    "Economics studies the production, distribution, and consumption of goods and services.",
    "The soccer match ended in a tie.",
    "Quantum physics explores the behavior of particles at atomic levels.",
    "In a market economy, supply and demand determine prices.",
    "Photosynthesis in plants was covered in biology class.",
    "The professor explained the Keynesian model of economics.",
    "We discussed the impacts of monetary policy on inflation.",
```

```
    "The recent layoffs in the company have left many employees feeling hopeless and
worried.",
    "The customer service was terrible, and I will never shop here again.",
    "The weather today is cloudy but not too cold, which makes it a neutral day for me."
]

# Load spaCy model for text preprocessing
nlp = spacy.load("en_core_web_sm")
processed_texts = [[token.lemma_ for token in nlp(text) if not token.is_stop and not
token.is_punct] for text in texts]

# Create dictionary and corpus for LDA
dictionary = corpora.Dictionary(processed_texts)
corpus = [dictionary.doc2bow(text) for text in processed_texts]

# Create LDA model
lda_model = LdaModel(corpus=corpus, num_topics=2, id2word=dictionary, passes=15)

# Get identified topics
topics = lda_model.print_topics(num_words=5)

# Create a transformers pipeline for sentiment analysis
sentiment_pipeline = pipeline("sentiment-analysis")

# Perform sentiment analysis on the texts
sentiment_results = []
for text in texts:
    sentiment_result = sentiment_pipeline(text)[0]
    sentiment_results.append((text, sentiment_result['label'], sentiment_result['score']))

# Display the results in a user-friendly way
print("=== Topic Analysis Results ===")
print("The topic analysis has identified the following predominant topics in the texts:")
for i, topic in topics:
    print(f"Topic {i+1}: {topic}")

print("\n=== Sentiment Analysis Results ===")
print("The sentiment analysis has evaluated the following texts:")
for text, label, score in sentiment_results:
    print(f"Text: {text}")
    print(f"Sentiment: {label.capitalize()} (Confidence: {score:.2f})\n")

# Generate word clouds for the topics
for i, topic in topics:
    words = {word.split('*')[1].replace('"', ''): float(word.split('*')[0]) for word in
topic.split(' + ')}
```

```
    wordcloud = WordCloud(width=800, height=400,
background_color='white').generate_from_frequencies(words)
    plt.figure(figsize=(10, 5))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis('off')
    plt.title(f"Word Cloud for Topic {i+1}")
    plt.show()

# Generate bar chart for sentiment analysis
sentiment_labels = [result[1] for result in sentiment_results]
positive_count = sentiment_labels.count('POSITIVE')
neutral_count = sentiment_labels.count('NEUTRAL')
negative_count = sentiment_labels.count('NEGATIVE')
plt.figure(figsize=(10, 5))
plt.bar(['Positive', 'Neutral', 'Negative'], [positive_count, neutral_count,
negative_count], color=['green', 'blue', 'red'])
plt.xlabel('Sentiment')
plt.ylabel('Number of Texts')
plt.title('Sentiment Distribution in Texts')
plt.show()
```

In the following section, the code is explained.

5.2 Explanation of the code

First, the code imports several libraries necessary for the analysis. spacy is used for text preprocessing, gensim for topic modeling, transformers for sentiment analysis, and matplotlib along with wordcloud for visualizations.
A list of sample texts covering various topics, including economics, sports, physics, and personal experiences, is defined.
The spacy model is loaded for text preprocessing. This step removes common words and punctuation marks, and converts words to their base forms (lemmas).
A dictionary is created that assigns unique identifiers to each word and a corpus that represents the texts as bags of words, where the frequency of each word is counted.
An LDA (Latent Dirichlet Allocation) model is trained to identify two topics in the texts.
The topics identified by the LDA model are extracted, showing the most representative words of each topic.
A transformers pipeline is created to analyze the sentiment of each text, classifying them as positive, negative, or neutral, along with confidence scores.
The results of the topic analysis are printed, showing the predominant topics identified in the texts.
The results of the sentiment analysis are printed, showing each text with its sentiment classification and confidence score.
Word clouds are generated for each identified topic, providing a visual representation of the key words associated with each topic.
A bar chart is generated showing the number of texts classified as positive, neutral, and negative, providing a clear view of the distribution of sentiments in the texts.

5.3 Result of the code execution

The result of the code execution is shown below.

=== Topic Analysis Results ===
The topic analysis has identified the following predominant topics in the texts:
Topic 1: 0.034"economic" + 0.021*"employee" + 0.021*"company" + 0.021*"layoff" + 0.021*"feel"*
Topic 2: 0.037"service" + 0.036*"discuss" + 0.036*"policy" + 0.036*"impact" + 0.036*"match"*

=== Sentiment Analysis Results ===
The sentiment analysis has evaluated the following texts:
Text: Economics studies the production, distribution, and consumption of goods and services.
Sentiment: Positive (Confidence: 0.99)

Text: The soccer match ended in a tie.
Sentiment: Positive (Confidence: 0.73)

Text: Quantum physics explores the behavior of particles at atomic levels.
Sentiment: Positive (Confidence: 1.00)

Text: In a market economy, supply and demand determine prices.
Sentiment: Positive (Confidence: 0.99)

Text: Photosynthesis in plants was covered in biology class.
Sentiment: Positive (Confidence: 0.66)

Text: The professor explained the Keynesian model of economics.
Sentiment: Positive (Confidence: 0.99)

Text: We discussed the impacts of monetary policy on inflation.
Sentiment: Positive (Confidence: 0.99)

Text: The recent layoffs in the company have left many employees feeling hopeless and worried.
Sentiment: Negative (Confidence: 1.00)

Text: The customer service was terrible, and I will never shop here again.
Sentiment: Negative (Confidence: 1.00)

Text: The weather today is cloudy but not too cold, which makes it a neutral day for me.
Sentiment: Positive (Confidence: 0.99)

Figure 2. Word Cloud for Topic 1 and Word Cloud for Topic 2

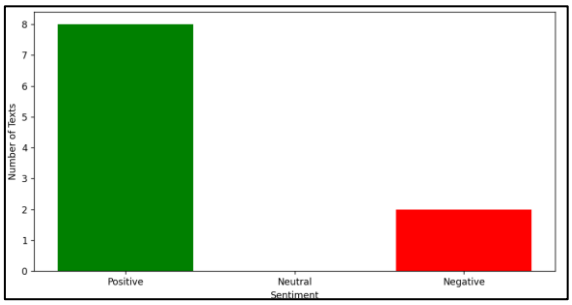

Figure 3. Sentiment Distribution in Texts

The topic analysis identified two main themes in the analyzed texts. The first theme focuses on economic and labor issues, with keywords like "economic," "employee," "company," "layoff," and "feel," suggesting concerns about job stability and economic impact. The second theme covers services, policies, and sports events, with keywords such as "service," "discuss," "policy," "impact," and "match."

Regarding sentiment analysis, most texts were classified as positive with high confidence, especially educational and informative texts on economics and physics. Texts describing negative events, such as layoffs and poor customer service, were classified as negative with high confidence. For example, "The recent layoffs in the company have left many employees feeling hopeless and worried" received a negative sentiment with a confidence of 1.00.

Overall, these results show the effectiveness of NLP models in identifying key topics and accurately assessing the sentiment in various texts.

Each topic identified by LDA is visualized using word clouds (Figure 2), highlighting the most important words. This visual representation helps to quickly understand the key elements of each topic.

The bar chart (Figure 3) shows the distribution of sentiments in the analyzed texts:

Positives: 8 texts, Neutrals: 0 texts, Negatives: 2 texts

These visual tools make the analysis results more accessible and interpretable, even for those without technical knowledge, providing an intuitive visual representation of the conducted analysis.

Conclusion

In this study, a methodology based on advanced NLP techniques for analyzing discussion forums in the field of economics has been presented. Utilizing the spaCy and Transformers libraries in Python, textual data has been systematically and robustly extracted and analyzed, providing valuable insights into student interactions and concerns.

This code provides a solid foundation for text analysis. However, to fully achieve the objective of identifying specific texts from economics forums, it is necessary to adapt and train the classification model with specific data from those forums. Additionally, integrating additional techniques such as topic analysis (LDA) would be advantageous to improve the detection of key terms and relevant topics. This requires collecting and labeling a representative dataset, as well as developing a more specialized classification model tailored to the needs of economic analysis.

The applied methodology, following an adapted CRISP-DM structure, allowed for efficient data collection and preprocessing, ensuring noise elimination and proper text preparation for analysis.

References

1. Ahmed, W., Bath, P. A., Sbaffi, L., & Demartini, G. (2022). Sentiment analysis in healthcare: A review of recent advances and future directions. Journal of Biomedical Informatics, 129, 104105. https://doi.org/10.1016/j.jbi.2022.104105
2. Aderibigbe, S. A., AbdelRahman, A. A., & Al Othman, H. (2023). Using Online Discussion Forums to Enhance and Document Students' Workplace Learning Experiences: A Semi-Private Emirati University's Context. Education Sciences, 13(5), 458. https://doi.org/10.3390/educsci13050458
3. Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. Journal of Data Warehousing, 5(4), 13-22.
4. Elsevier. (n.d.). Scopus: Access and use support center. Retrieved from https://www.elsevier.com/solutions/scopus
5. Saravanan, K. S., & Bhagavathiappan, V. (2024). Innovative agricultural ontology construction using NLP methodologies and graph neural network.
6. Júnior, F. S., Reis, P. A., Cavalcante, M. S., & De Oliveira, A. H. M. (2024). Systems Engineering Process Enhancement: Requirements Verification Methodology using Natural Language Processing (NLP) for Automotive Industry.
7. Seong, N. K., Lee, J. H., Lee, J. B., & Seong, P. H. (2023). Retrieval methodology for similar NPP LCO cases based on domain specific NLP.
8. Bellandi, V., Maghool, S., & Siccardi, S. (2023). An NLP-based statistical reporting methodology applied to court decisions.
9. Biswas, B., Sengupta, P., Kumar, A., Delen, D., & Gupta, S. (2022). A critical assessment of consumer reviews: A hybrid NLP-based methodology.

10. Goulas, A., Malamas, N., & Symeonidis, A. L. (2022). A Methodology for Enabling NLP Capabilities on Edge and Low-Resource Devices.
11. Ramos-Gutiérrez, B., Varela-Vaca, Á. J., Ortega, F. J., Gómez-López, M. T., & Wynn, M. T. (2021). A NLP-Oriented Methodology to Enhance Event Log Quality.
12. De, T., & Mukherjee, D. (2021). Explainable NLP: A Novel Methodology to Generate Human-Interpretable Explanation for Semantic Text Similarity.
13. Wang, Y. (2020). Basic Methodologies Used in NLP Area.
14. Qiu, M., Housh, M., & Ostfeld, A. (2020). A two-stage LP-NLP methodology for the least-cost design and operation of water distribution systems.
15. Castillo-Zúñiga, I., Luna-Rosas, F. J., Rodríguez-Martínez, L. C., Muñoz-Arteaga, J., López-Veyna, J. I., & Rodríguez-Díaz, M. A. (2020). Internet data analysis methodology for cyberterrorism vocabulary detection, combining techniques of big data analytics, NLP and semantic web.
16. Amato, F., Cozzolino, G., Moscato, V., & Moscato, F. (2019). Analyse digital forensic evidences through a semantic-based methodology and NLP techniques.
17. Moreno, D. C., & Vargas-Lombardo, M. (2018). Design and construction of a NLP based knowledge extraction methodology in the medical domain applied to clinical information.
18. Juanals, B., & Minel, J. L. (2018). An instrumented methodology to analyze and categorize information flows on twitter using nlp and deep learning: A use case on air quality.
19. Vargas, V. M. C., Stephens, C. R., Martínez, G. E. S., & Rendón, A. M. (2015). Nlp methodology as guidance and verification of the data mining of survey ensanut 2012.
20. Niu, J., & Issa, R. R. A. (2014). Rule-based NLP methodology for semantic interpretation of impact factors for construction claim cases.
21. Hadi-Vencheh, A., & Mohamadghasemi, A. (2013). An integrated AHP-NLP methodology for facility layout design.
22. Mills, M. T., & Bourbakis, N. G. (2012). A comparative survey on NLP/U methodologies for processing multi-documents.
23. Kovačević, A., Konjović, Z., Milosavljević, B., & Nenadic, G. (2012). Mining methodologies from NLP publications: A case study in automatic terminology recognition.
24. Irwin, J. Y., Harkema, H., Christensen, L. M., Schleyer, T., Haug, P. J., & Chapman, W. W. (2009). Methodology to develop and evaluate a semantic representation for NLP.
25. Heinze, D. T., Morsch, M. L., Potter, B. C., & Sheffer Jr., R. E. (2008). Medical i2b2 NLP Smoking Challenge: The A-Life System Architecture and Methodology.
26. Castillo-Zúñiga, I., Luna-Rosas, F. J., Rodríguez-Martínez, L. C., Muñoz-Arteaga, J., López-Veyna, J. I., & Rodríguez-Díaz, M. A. (2020). Internet data analysis methodology for cyberterrorism vocabulary detection, combining techniques of big data analytics, NLP and semantic web.
27. Juanals, B., & Minel, J. L. (2018). An instrumented methodology to analyze and categorize information flows on Twitter using NLP and deep learning: A use case on air quality.
28. Jurafsky, D., & Martin, J. H. (2019). Speech and Language Processing (3rd ed.). Prentice Hall. Retrieved from https://web.stanford.edu/~jurafsky/slp3/
29. Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media.
30. Müller, A. C., & Guido, S. (2016). Introduction to Machine Learning with Python: A Guide for Data Scientists. O'Reilly Media.

31. Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media.
32. Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. Retrieved from https://spacy.io
33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (pp. 5998-6008).
34. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (pp. 38-45).
35. Řehůřek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks (pp. 45-50).
36. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). CRISP-DM 1.0 Step-by-step data mining guide. Retrieved from https://www.the-modeling-agency.com/crisp-dm.pdf