Applied Computer Science, vol. 20, no. 2, pp. 24–42
doi: 10.35784/acs-2024-14

Submitted: 2024-01-21 | Revised: 2024-04-08 | Accepted: 2024-04-14

Keywords: artificial intelligence, digital news, machine learning, text mining

Fernando Andrés CEVALLOS SALAS [0009-0002-5222-2599]*

DIGITAL NEWS CLASSIFICATION AND PUNCTUACTION USING MACHINE LEARNING AND TEXT MINING TECHNIQUES

Abstract

Persistent growth of information in recent decades, along with the development of new information technologies for its management, have made it essential to develop systems that allow to synthesize this massive information or better known as big data. In this article, a feedback based system for massive processing of digital newspapers is presented. This system synthesizes the most relevant information from different news stories obtained from several sources. System is fed with information from the Internet using web scraping techniques. All this information is stored in a data lake which has been implemented using NoSQL databases. Next, data processing is performed, focusing on words, their relevance, and their correlation with other words from related content groups or headlines. In order to perform this aggrupation, machine learning Large Language Model (LLM), K Nearest Neighbors (KNN) and text mining techniques are used. New text mining algorithms are also developed to adjust thresholds during content aggregation and synthesis. Finally, the results visualization mechanism is presented which allow users to give a punctuation to the news stories. This mechanism represents a feedback punctuation for the system which will be considered into the global punctuation, which is the basis to show the results. This system can be useful to summarize all the information contained in the news stories which are stored in Internet, providing users a fast way to be informed.

1. INTRODUCTION

Increasing data volume has led to the development of data science tools and frameworks (Kannan et al., 2016). In this context, it is essential to develop new processes and algorithms that leverage these tools to efficiently and effectively process data.

It is estimated that the volume of data in the world should reach 163 zettabytes by 2025 and that the average person will interact with electronic devices 4,800 times a day in that same year (Ribeiro, 2019). Combination of these two factors, the exponential growth of data and the growing digital interaction, generates the need for systems that can synthesize the information from different sources that have a significant relationship in a practical way.

* Escuela Politécnica Nacional, Departamento de Informática y Ciencias de la Computación, Ecuador, fernando.cevallos03@epn.edu.ec

Information should be structured, indexed, and easily accessible to the end user (Bustamante & Guillén, 2020). Therefore, it is important to focus on building systems which help users to access information in an interactive way.

Currently proliferation of newspapers (institutions) in some countries has reached the figure of 50. Emergence of the Internet has allowed readers to access these sources in electronic format freely. However, the synthesis of information from this countless number of sources is overwhelming for any reader.

Text mining is a general concept of data mining that aims to automatically extract knowledge from unstructured data sources (Berry & Kogan, 2010). Unstructured data are those that do not have a defined format or schema (Balusamy et al., 2021). Text mining plays an important role in many application areas such as economy, social administration, information services, and security (Zong et al., 2021).

Machine Learning is part of Artificial Intelligence which main goal is to enable computer systems to learn from data, in a way that emulates human capabilities (Bobadilla, 2021).

This article presents the structure of an integrated text mining and machine learning system for the synthesis of information from digital newspapers. This system uses web scraping techniques to collect information from the main digital newspaper sources. Subsequently, the information is processed to offer the end user a content synthesis service. Main purpose of the system is to offer the user the most relevant information in a quick and concise way. In order to achieve this, the system uses an algorithm that identifies the most important content of each news story and presents it in an information tab. This system is useful for people who do not have time to read the newspaper, or several sources of digital newspapers, but want to stay up-to-date on the latest current events.

Section 2 describes the machine learning models which were chosen to develop the system. Section 3 is an overview about the methodology selected. Section 4 details data collection, including intake and storage. Sections 5 and 6 describe data processing. Section 7 details access to the results, which is reached through a web interface. Section 8 focuses on feedback, which allows the system to improve its effectiveness. Section 9 it's about the execution and result metrics. Section 10 describes the conclusions, which highlight the potential benefits of the system.

2. MACHINE LEARNING MODELS

Large Language Models have rapidly risen to prominence as one of the most promising and transformative technologies in artificial intelligence (Almeida, 2023). Large Language models like OpenAI's, GPT-4, BERT, Falcon, etc., are the artificial intelligence models that can understand and generate text in human languages. They have been trained on huge volumes of text and code to predict the next token of text (Koul, 2023).

KNN is a supervised machine learning algorithm focused on the classification and regression of data samples (Johri et al., 2020). This algorithm measures the average measurements of adjacent nodes and estimates the missing values (Suganthi et al., 2021). KNN performs computation based on Euclidean distance to know which are the nearest neighbors.

Machine Learning models which have been used in order to develop this system are BERT Large Language Model (BERT LLM) and K Nearest Neighbors (KNN).

2.1 Bert large language model

Bidirectional Encoder Representations from Transformers (BERT) LLM is a multilayer bidirectional transformer encoder. BERT has been designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers (Bao et al., 2023). BERT has been developed by Google and constitutes an LLM build on artificial neural network.

The main characteristic of BERT LLM is its bidirectionality, which allows this LLM to a better understanding of the meaning of a word among those around it. It is why BERT LLM has been chosen to accomplish this system.

BERT allows to transform words and also complete sentences into vectors (Campesato, 2023). BERT vectors are created focusing on cosine similarity (Chen et al., 2023).

Once the BERT vectors are obtained, they can be used to perform computational operations on them. The conversion from word to vector increases the efficiency of operations while allowing words to be interrelated by cosine similarity.

2.2 K nearest neighbors model

KNN is a non-parametric supervised learning technique developed in 1951 by Evelyn Fix and Joseph Hodges. KNN is also known as a lazy learning algorithm due to no training data is required. When an instance of an unknown class is presented for evaluation, the algorithm computes its K closets neighbors (Rajaguru & Prabhakar, 2017).

KNN has been widely used in classification problems (Wang et al., 2006). Vectors generated by the BERT algorithm, by storing cosine relationships, can be related based on the KNN algorithm. This allows these two algorithms to be used as a highly effective search engine which will get the related BERT vectors which correspond to text. Also, this search engine will allow the final users to perform semantic search.

Semantic search is a set of search engine capabilities, which main goal is to understand the words searched and their context (Amerland, 2013).

3. METHODOLOGY

Cross Industry Standard Process for Data Mining (CRISP-DM) has been used to develop this system. The CRISP-DM methodology is a multinational, standards-based approach to describe, document and continuously improve data mining (and associated data warehousing, business intelligence) processes (De Ville, 2001). CRISP-DM was one of the first industrial data mining and knowledge process models. It was developed in late 1996 by a large consortium of European companies (Rahman El Sheikh & Alnoukari, 2012).

CRISP-DM Methodology is composed by six phases: Business understanding, Data understanding, Data preparation, Modeling, Evaluation, and Deployment (Yale et al., 2017). The order of the phases is not strict since the results of one phase may show that more effort is required in a previous one (Hildebrandt & Gutwirth, 2008). The first and second phase consist of performing an understanding, the third phase prepares the data to be used in the system, the fourth and fifth phase describe and execute the steps for the operational solution, finally the last phase deploys and communicates the results (Abramowicz & Tolksdorf, 2010).

The CRISP-DM methodology is widely used in data mining projects, as it provides a clear idea of the structure of the project's life cycle (Sánchez Trujillo & Pérez Hernández, 2021), while allowing it to go back between phases. This allows the project to gradually achieve the desired objectives.

4. DATA COLLECTION

Data collection for the system is carried out using web scrapers which analyze and download information from various digital newspaper sources. Source diversity helps for a global view of the story events (Gorelik, 2019).

Main goal of the data collection system is to store raw data in a data lake. A data lake is a repository that stores data from various sources in its original format, unprocessed or with minimal processing (Pasupuleti & Purra, 2015). This allows organizations to perform subsequent analysis according to their needs (Gils, 2023).

To achieve this goal, the system's data lake is primarily implemented using a NoSQL database, MongoDB. MongoDB stores information in BSON documents, a binary data format. The main difference with relational databases is that MongoDB stores documents, not rows.

To perform data ingestion, the information from digital newspapers will be stored into the data lake using web scrapers which extract the news stories from the Internet and store each one as a separated document in the MongoDB database. Figure 1 shows the architecture used to data collection.

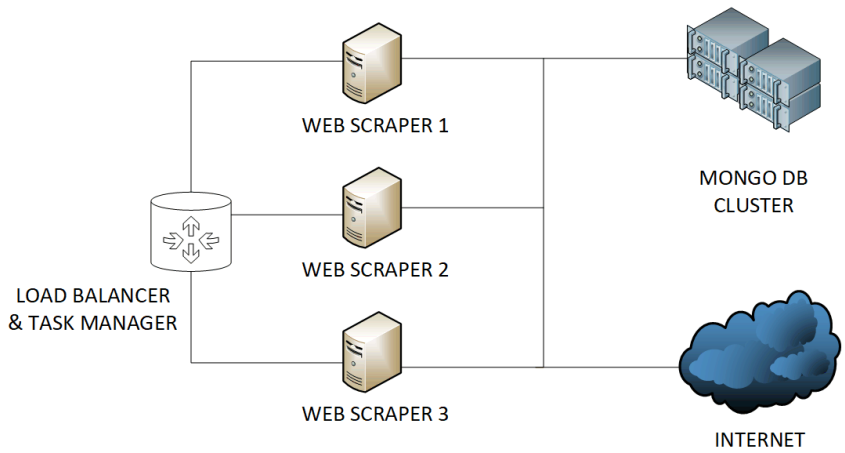

Fig. 1. Massive data ingestion architecture

Stored information in the MongoDB database must be pre-processed to standardize the different formats in which it is collected from the different sources. Therefore, four fields are considered for storage purposes. Figure 2 shows these fields. For each news story from each newspaper, its headline and full text content are considered. Text content is also separated line by line to facilitate the analysis and query of specific lines later. These lines are stored to reduce processing times in the following phases. In addition, a field with a

unique code auto-generated by the ingestor service is included to uniquely identify the news article in subsequent procedures.

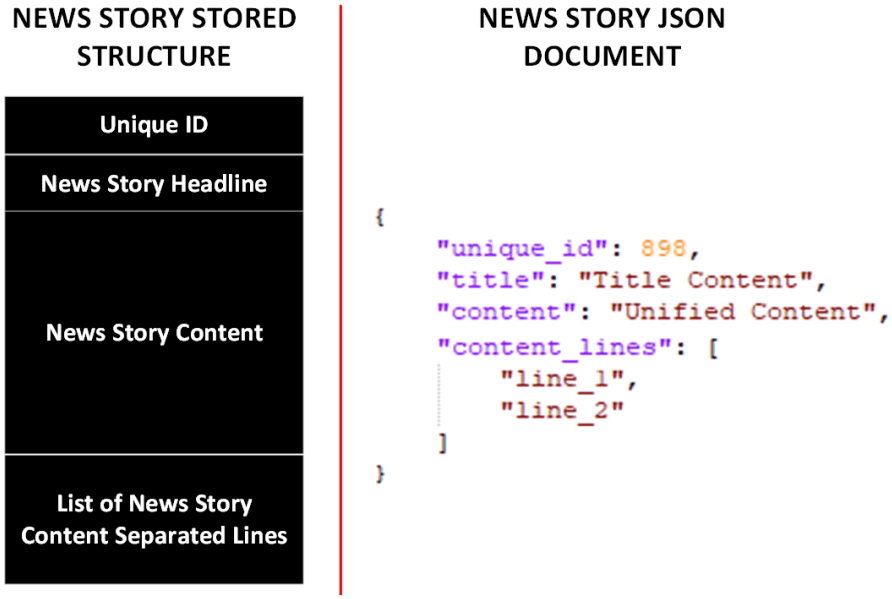

Fig. 2. News story structure to be stored in data lake

5. DIGITAL NEWS STORIES CLASSIFICATION ALGORITHM

Digital news stories grouping by topic is a necessary task for data analysis, due to the information is collected from various sources and each has different writing styles.

After storage the news stories in the data lake, the information corresponding to the headline and content lines of each one is stored encoded by BERT LLM that allows for semantic searches. Semantic searches allow you to find results that are similar to the query, even if they do not match exactly. For example, if you search for the word "traditions", a semantic search will also return the lines that contain "traditional".

In order to perform this kind of searches, BERT LLM has been used to transform headlines and content lines of the news stories into vectors. Each headline or content line is transformed into a 768 vector dimension which will be stored in an indexed database.

To perform a search between vectors it has been implemented KNN model which will search the nearest neighbors, these neighbors correspond to the headlines which are similar between them.

In order to collect the similar stories (neighbors) guaranteeing correspondence between them, a clustering algorithm can be implemented that identifies the similarity between news headlines. This algorithm will allow news stories to be grouped by headline similitude, which will facilitate subsequent analysis phases.

Figure 3 shows the flow chart for clustering digital news stories. This algorithm works in conjunction with BERT LLM and KNN algorithms, using BERT LLM to convert words to vectors and KNN as a search engine to perform semantic searches. Grouping algorithm uses

the headline stored in the previous step (data collection) to compare each news story with the others. The goal is to group news stories from different newspapers by topic.

The results of this processing phase will be stored in a database. This database also contains a list of stop words which do not contribute to the search process and should be removed from queries to the search engine. These stop words include, for example, "in", "the", "about", "for", among others.

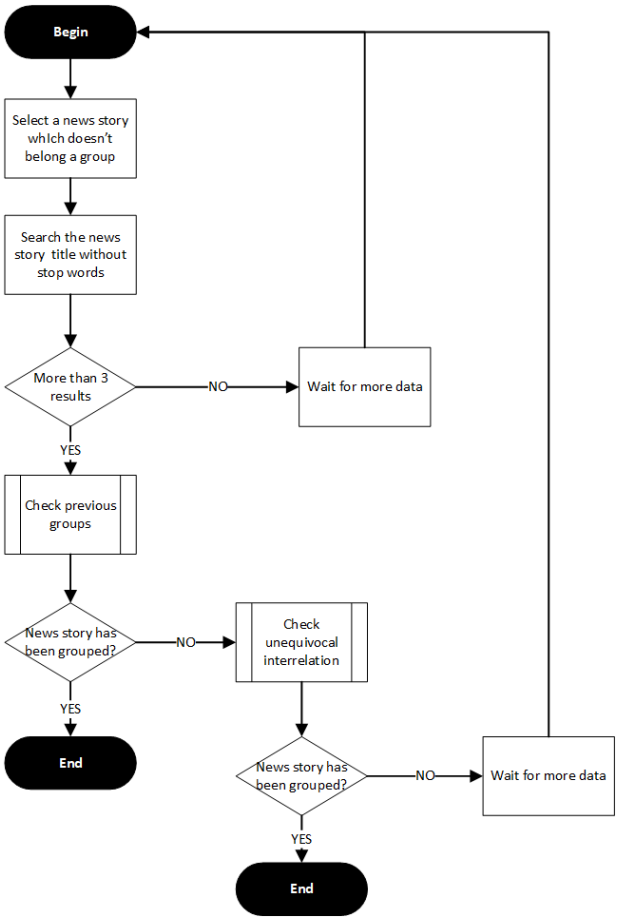

Fig. 3. Digital news stories aggrupation algorithm flow chart

Aggrupation algorithm begins taking a news story collected which is going to be grouped. News story selected (root news story) will be used to query using the search engine. Algorithm removes the stop words from the headline and performs a search for similar headlines in the search engine. The search must have at least three results (resulting news stories). If there are fewer than three results, the root news story is discarded because there are not enough news stories to group it and it waits until the data collection process provides more similar news stories to perform the grouping. If there are three or more results, the algorithm continues. The first three headlines of other news stories that matched are taken. Apparition order in which the search engine returned the results is important.

First three results obtained are verified in order to check if they already belong to a previous group. If so, a weighted assessment is carried out based on the order of the results.

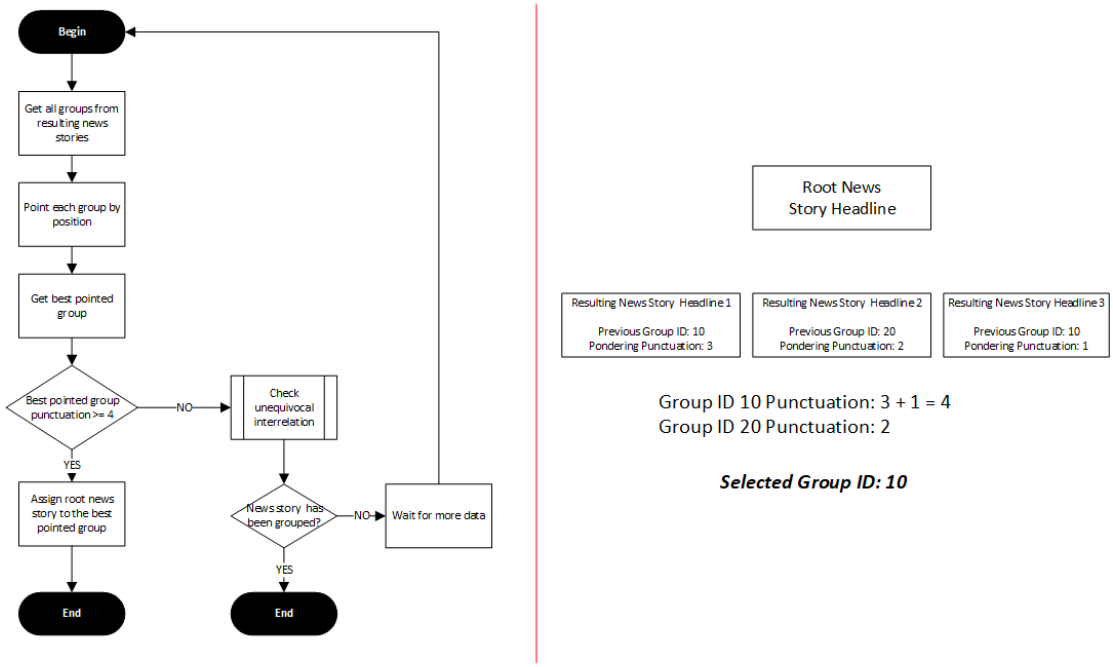

Fig. 4. Check previous groups phase flow chart and example

Figure 4 shows the algorithm for weighting previous groups and an example. It can be observed that the first result contributes with a weight of 3, the second with a weight of 2 and the third with a weight of 1. Accumulated result can be quantified for each of the groups involved in the first three results. The group with the highest number of points, as long as this number is equal to or greater than 4, will be selected and the root news story will be integrated into the selected group.

There is a particular case in which the three resulting news stories belong to different groups and the minimum score of 4 points cannot be reached. In this case, the root news story is discarded until the data collection process provides news stories with greater interrelatedness. This is because the root news story has too general (superficial) content and does not significantly contribute to the study of a specific group, but contributes superficially to several.

Sometimes, there is not enough information to integrate a news story directly, based on the weighting of previous groups. This can happen, for example, if one or more of the news stories obtained in the search have not been assigned to a group. Figure 5 shows a case in which a result of this type was obtained. In this case, the unclassified result is in first place and is crucial, since the minimum 4 points required in the weighting of previous groups cannot be obtained.

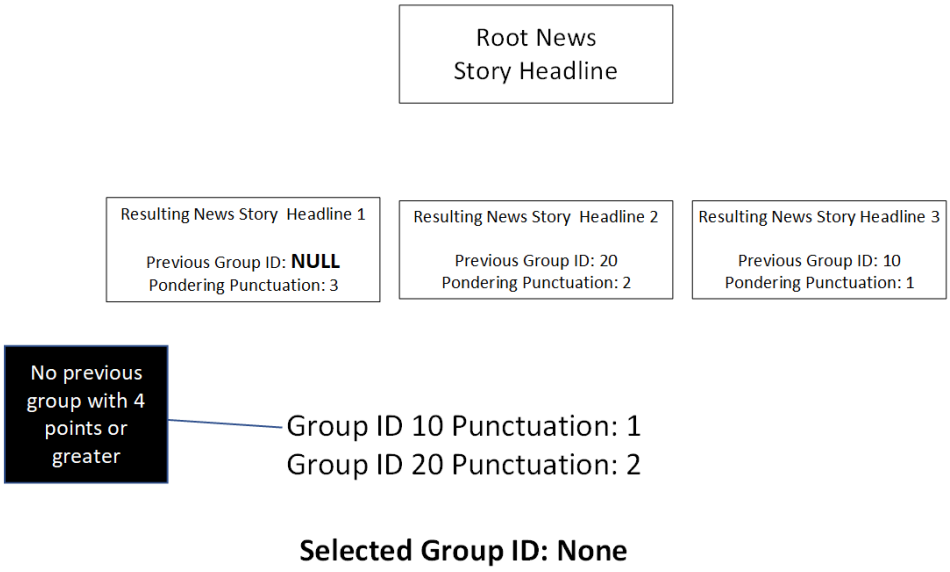

Fig. 5. Resulting news stories with and without previous group

In order to address cases of news without a previous group, whether one or several, the unequivocal interrelation algorithm is used. This algorithm aims to measure the degree of interrelation between the root news story and the other three resulting news stories. Figure 6 describes this algorithm phase. Initially, the search corresponding to the three results is performed one by one. For example, for the resulting news story 1, it becomes temporally in a "root news story" and the search for the news story headline 1 is performed using the search engine. If the original root news story headline appears among the results given by the search engine, a relationship can be established between news story 1 and the original root news story. The same procedure is followed with news story 2 and 3. However, there are certain considerations to take into account. When searching using the resulting news stories, the original root news story headline must be within 70% of the first results or within 70% of the number of sources involved in the system data collection. The algorithm uses the minimum value. For example, if after searching for resulting news story 1, the search engine has returned 20 results, there is a limit of 14, corresponding to 70% of the results. It should be considered that if data collection is being performed from 10 sources, there is a limit of 7, corresponding to 70% of the number of sources. Since the smallest value is 7 to establish the unequivocal interrelation, then the original result must be within the top 7. In this way, a similarity concordance can be established between the two stories.

After establishing which news stories correspond to or are unequivocally interrelated with the root news story, they can be grouped together. If there is a news story that already has a previous group and has passed the unequivocal interrelation algorithm, the group of the news story with the best position in the result will be taken, as shown in Figure 7. This group will be associated with all news stories that passed the unequivocally interrelation process which do not have a group and also with the root news story. In Figure 7, news story 2 has a previous group and is in the best position, so the news stories will be associated with

this group. On the other hand, news story 3, which already is part of a previous group, remains intact, as it belongs to a different previous grouping.

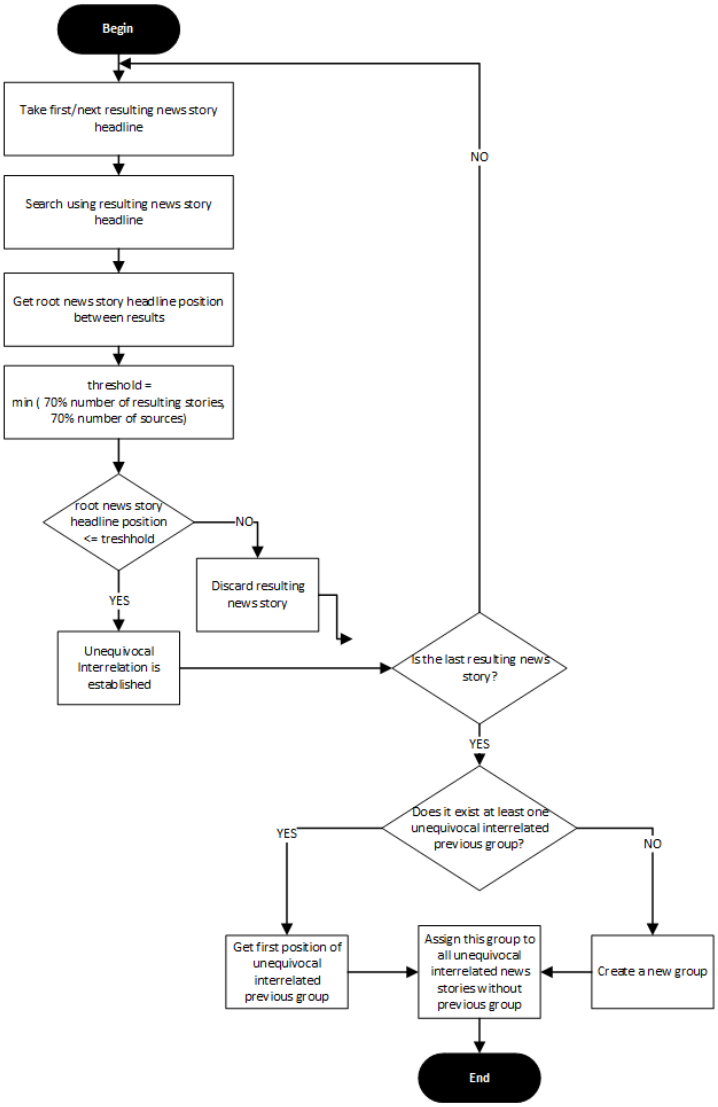

Fig. 6. Check unequivocal interrelation phase flow chart

Unequivocal interrelation algorithm is based on the bidirectional similarity that occurs within the defined thresholds when performing the bidirectional search of the news headlines. Unequivocal interrelation allows grouping news stories which do not have a group, including the root news story, and at the same time allows reaffirming the predominance of previous groups in news stories that have passed the algorithm and are part of already defined groups.

It is worth noting that the initial source for performing groupings is through the unequivocal interrelation algorithm. On the other hand, the previous group phase allows to improve the efficiency of the classification algorithm.

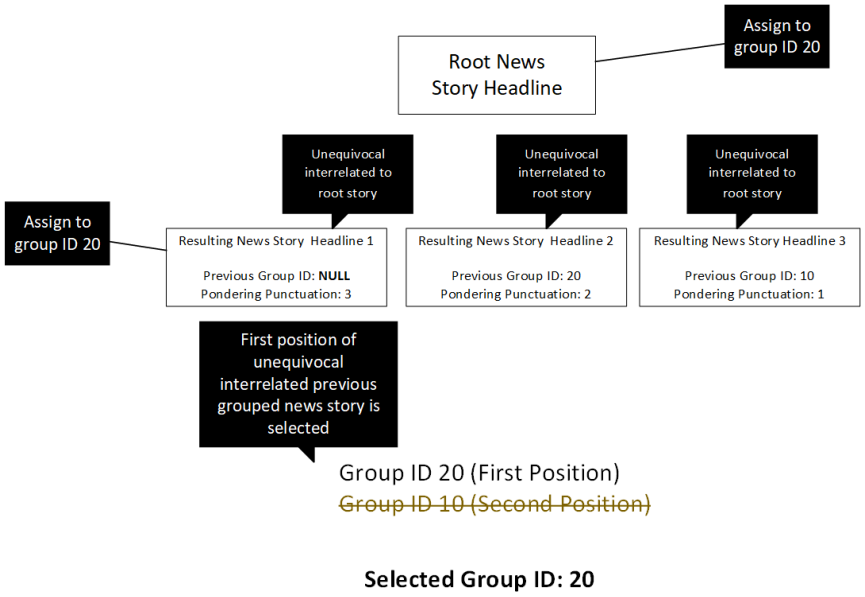

Fig. 7. Grouping process with several types of previous groups

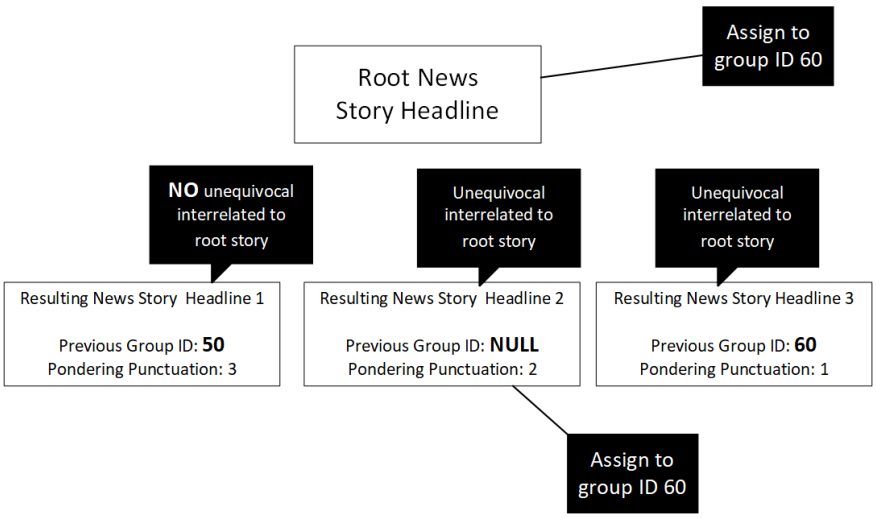

Fig. 8. News stories which did not pass the unequivocal interrelation algorithm successfully

It is possible that a news story does not successfully pass the unequivocal interrelation algorithm. In this case, that news story will not be taken into account to define the grouping.

Figure 8 shows a hypothetical case in which news story 1 did not pass the algorithm and a bidirectional relationship cannot be established between the root news story and news story 1. In this case, news story 1 will remain intact and the group value to be used will be that of news story 3, which did pass the algorithm.

When none of the news stories that have passed the unequivocal interrelation algorithm has a previous group, a new group is created. At the same time, the system will assign a unique group identifier.

6. CONTENT COMBINATORIAL REDUCTION ALGORITHM

After grouping, it is necessary to combine and reduce the content of the news stories belonging to a particular group. This is achieved by using the content combinatorial reduction algorithm. BERT encoded text of stored messages is used. The content of the first news story in the group is taken and separated by lines. For each line, stop words are removed. For each remaining word, the number of lines containing that word in each message belonging to the group is searched for. The cosine similarity percentage results are averaged to obtain the word score.

Figure 9 shows the content combinatorial reduction algorithm and an example in which a line from one of the news stories is scored based on the comparison of the similarity of its words with the other news stories. After being compared with the other news stories within the group, a score has been obtained. The average of the scores per word will allow us to know the total result per line.

Content combinatorial reduction algorithm scores each line of the stories which belong to the group. The scored lines are stored in descending order based on their resulting score; this produces a combination of lines between the several sources. Subsequently, the 7 lines with the highest score in the group are taken. In this way, the lines of greatest relevance of the digital newspapers corresponding to the same group have been filtered. The text has been reduced and combined.

Content combinatorial reduction algorithm is based on the premise that the words of greatest importance have a higher degree of appearance in the different sources of information. This increases the probability of truthfulness because it confirms the facts declared in the sources. In other words, this information has a higher probability of being true because it appears in several sources.

Results corresponding to all the scores are stored in the relational database, identified by the unique news story, group, and line codes. Figure 10 shows the storage structure for a scored line within the relational database. In this database, the score of each line and the fields for the feedback mechanism, described in Section 8, must be stored.

Combined and reduced results can be shown to the end user. The textual content of each line can be retrieved from the data lake at high speed, since during the data collection, the news stories codes and their separated lines were stored. Unique codes are used as references in the relational database.

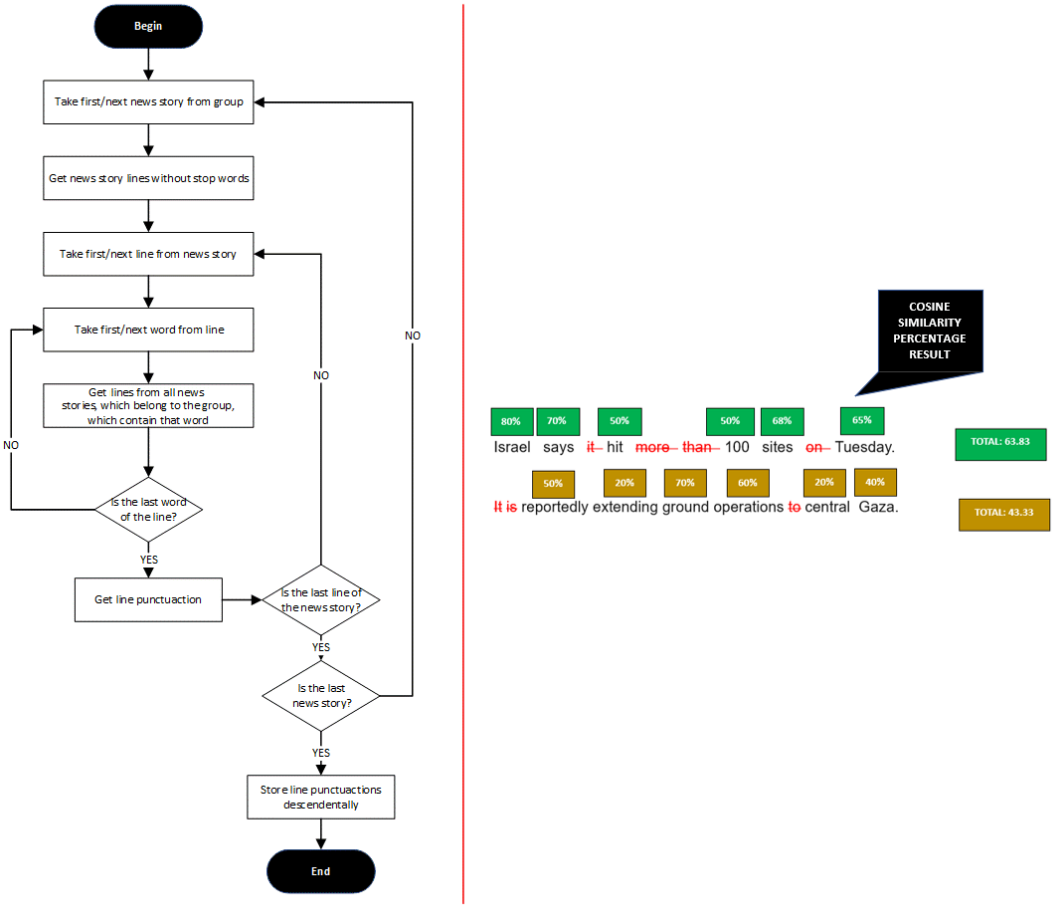

Fig. 9. Content combinatorial reduction algorithm flow chart and example

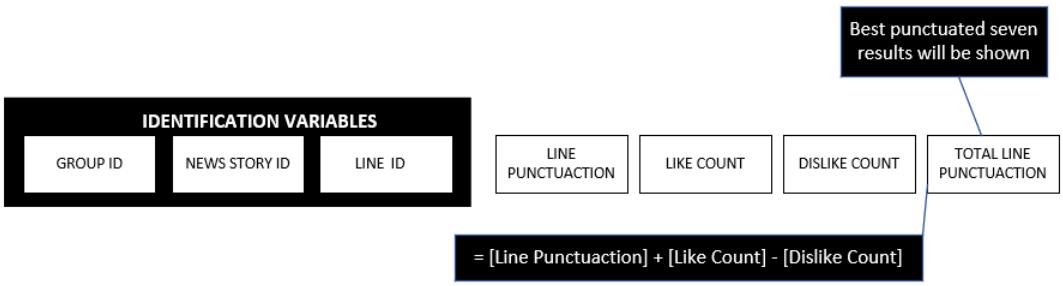

Fig. 10. Storage structure of score for each line of news stories

7. USER INTERFACE

Results can be displayed using a user interface that allows users to use the search engine. User interface allows users to perform semantic searches to obtain the results of the processed news stories. Results are sorted by search score.

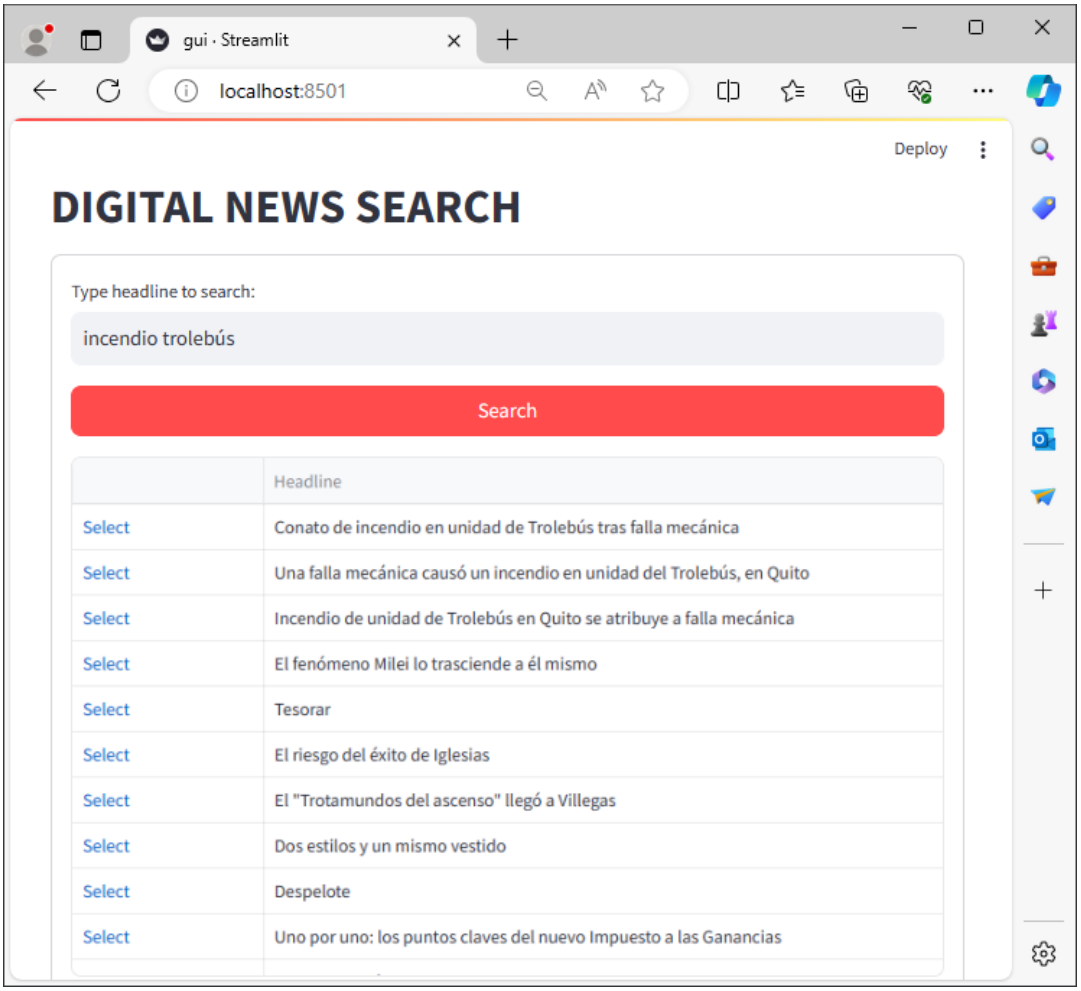

Fig. 11. User interface home screen

After performing a search, the results will be shown and when the end user opens one of the results, the synthesized information that has been obtained after data processing will be shown. This information is displayed as an informative tab, which allows the user to be informed quickly and easily, without having to read all the digital newspapers from several sources.

User interface structure is described in Figure 11. On the home screen, the user can access the search engine to query a topic. The search engine will respond with the news headline that most closely matches the query.

8. CONTENT FEEDBACK BY SCORE

On home screen when clicking on a result, the informative tab will be displayed with the content that is relevant to the reader, as shown in Figure 12.

Users can rate the results from the application, giving a "like" or a "dislike" to each of the 7 filtered lines, depending on the contribution that this content has provided them. The votes add or subtract one point to the total score per line that was stored in the relational database, as shown in Figure 10.

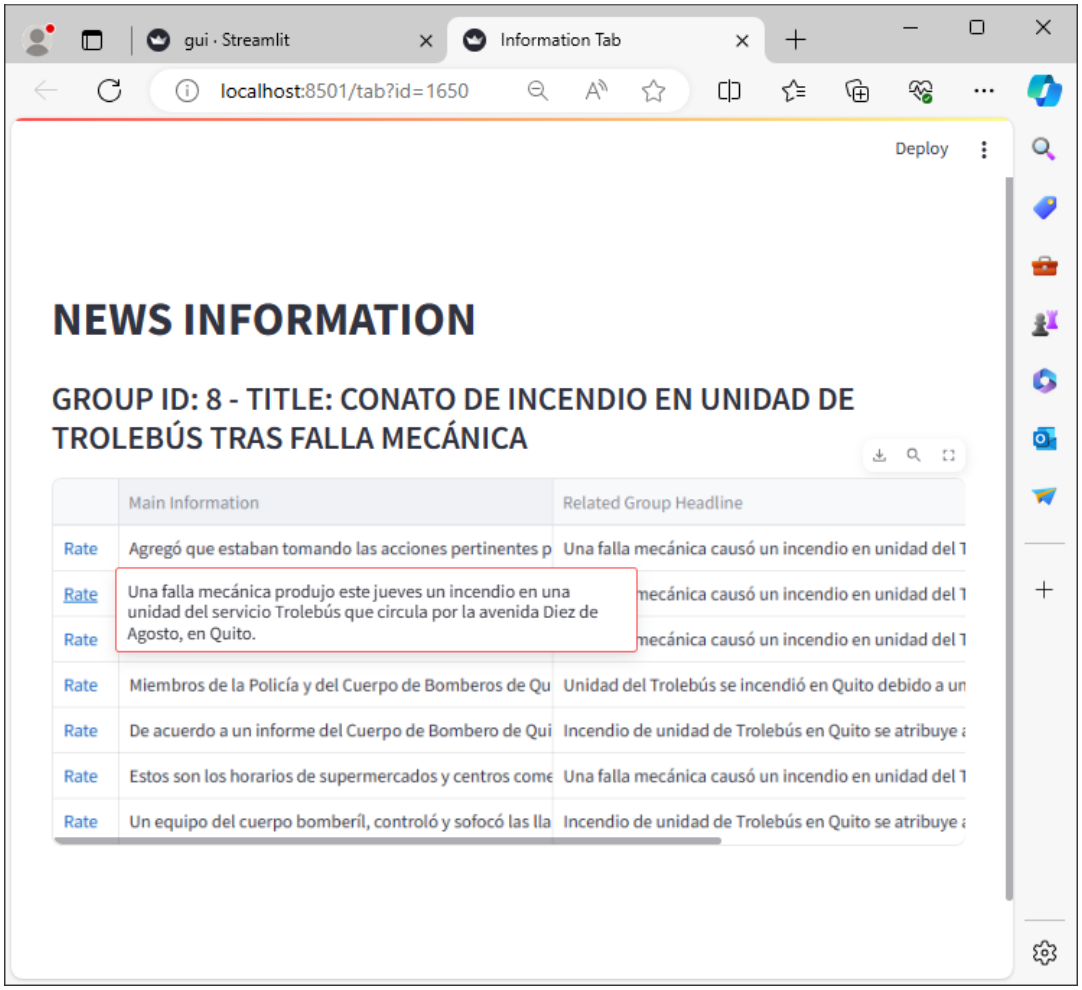

Fig. 12. User interface information tab screen

As shown in Figure 12, end users can select the line which want to rate. Next a window will be shown which will allow users to rate the selected content. Figure 13 shows the user interface feedback screen in which content that has been synthesized can be rated. User opinions can be positive (“like”) or negative (“dislike”); and influence the order of the results, so that a line can decline in scoring and stop being displayed on the platform because it is no longer among the first 7 results, allowing another better scored line to be displayed.

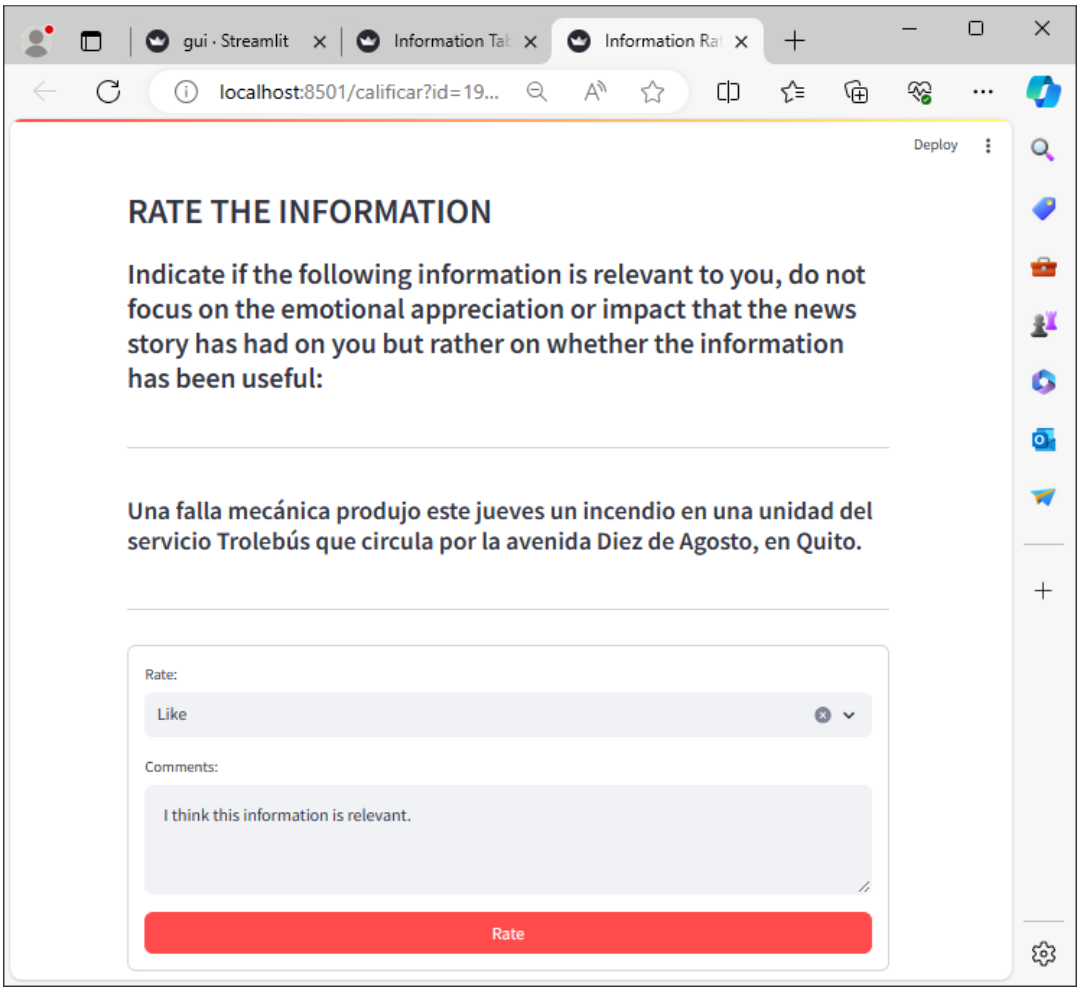

Fig. 13. User interface feedback screen

9. EXECUTION AND RESULT METRICS

For the execution process 49,630 digital news stories were processed. These news stories were collected during the web scraping process. Content of these news stories was separated into 537,146 lines.

System was implemented using Anaconda 3 framework with Python as programming language. PostgreSQL relational database engine was chosen for structured data storage. Elasticsearch engine, known for its efficiency, was used for fast search information retrieval. Finally, a MongoDB database engine was employed for handling web scraped data. Databases and search engine components were implemented in a server for data storage and separated from three processing nodes to handle the computational demands. The specific details of the implementation are illustrated in Figure 14.

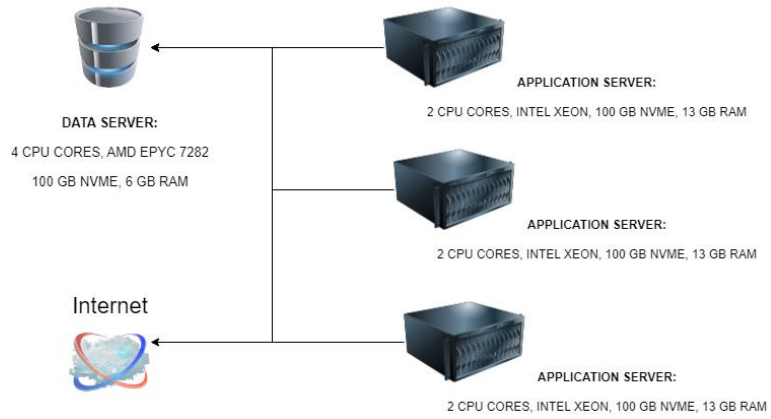

Fig. 14. System setup

After obtaining the results, the individual score of each line was averaged to obtain the score per news story. Process resulting news story score value can be contrasted based on other existing metrics. The Page Authority ranking of each news story was used for comparison. This metric is used as a reference to contrast the effectiveness of the model based on the semantic content process followed. Page Authority is a rank score based on an integer number between 0 to 100. It is calculated based on parameters which include mozRank, mozTrust, linking root domains, total number of referenced links, among others. Page Authority is calculated for a single page using a logarithmic scale in order to get one score (Kumar, 2020).

The estimation and Page Authority metrics were classified into 10 classes or categories, considering a range of 10 points for each class. These values which represent the quality of the news will be compared. Figure 14 shows the class distribution which has been used.

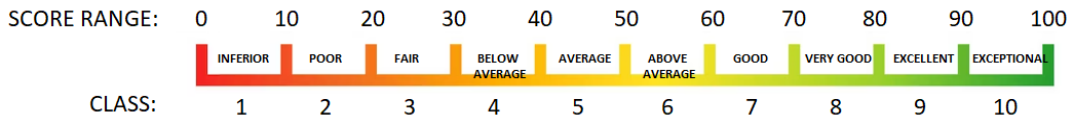

Fig. 14. Score range classes to classify news score

Based on the obtained classes, a comparison is made, this allows to appreciate the effectiveness of the built model. Figure 15 shows the metrics obtained for each of the defined classes.

Average accuracy is 85%, while for precision is 86%, the average f1-score is 85% and the average recall is 85%. Obtained results, as well as the clusters assigned, headlines and their contents can be downloaded in CSV format from (Cevallos, 2024). Table 1 shows the parameters of the resulting dataset for the headlines and content of each news story.

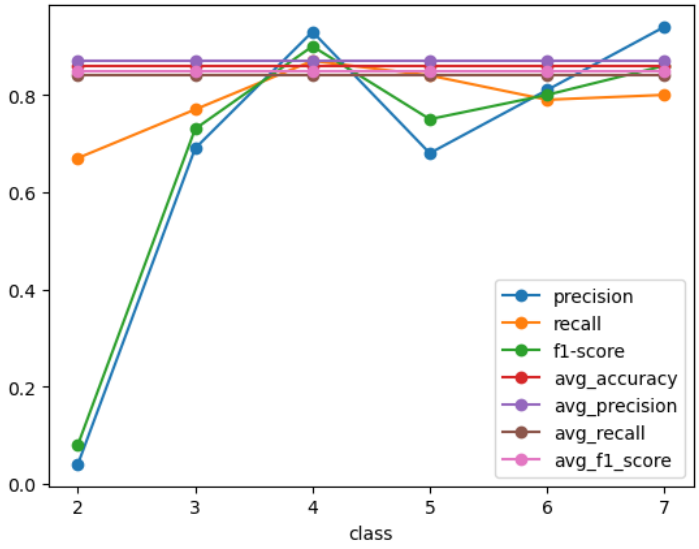

Fig. 15. Comparative metrics for each class

Tab. 1. Dataset parameters containing headlines, news content and results

News Headlines	
Parameter	**Description**
id	Unique id of headline
title	Headline title
title_without_stop_words	Headline title after stop words have been removed
page_authority	Page authority score
page_authority_category	Page authority score categorized from 1 to 10
calculated_score	Estimated score
calculated_category	Estimated score categorized from 1 to 10
cluster_id	Group ID to which belongs the news story
News Content	
Parameter	**Description**
id	Unique id of line content
id_headline	Unique id of headline to which belongs the content
line_content	Content
line_without_stop_words	Line content after stop words have been removed
score	Estimated score

10. CONCLUSIONS

Text mining has gained increasing importance in recent years due to the large amounts of text that are created on the Internet daily (Aggarwal & Zhai, 2012). Text classification processes and algorithms are becoming increasingly necessary.

Through the process described, it has been possible to cluster digital newspapers and then obtain the most relevant information. The clustering process, which involves assigning text to different groups or categories (Zong et al., 2021), is achieved based on an unequivocal

interrelation algorithm that correlates news headlines and is refined in performance using a previous groups algorithm. Subsequently, the content is treated using a combinatorial reduction algorithm, which allows for the synthesis of the text.

Text mining is useful to extract relevant information. Nowadays, it is impossible for a user to read all the news sources that are published on the Internet. Presented system allows user to get relevant information by synthesizing similar news stories. In addition, it is important to take into account user feedback, which was achieved with a binary rating system (like, dislike). Data storage, processing and visualization tools play a fundamental role in data analysis. These tools have allowed us to implement analytical processes, as well as the ingestion and construction of data lakes for storage. However, the adoption of different information technologies comes with a number of technical challenges that a data analyst must overcome to achieve his goals (Ortega, 2022). In this sense, it is important to select technologies assertively, taking into account the specific needs of each case. For example, NoSQL databases are often used to build data lakes, as they are more flexible and scalable than relational databases. Relational databases, on the other hand, are more suitable for storing processed information that can be structured and indexed. It can be stated that technologies must go hand in hand with data analysis strategies (Rúa Pérez, 2009). It is necessary for the data analyst to have a good understanding of the different available technologies and their capabilities.

This system can be integrated with other artificial intelligence, machine learning, or deep learning components. System resulting information can be used as input for other algorithms. Because this information has been processed and contains only the most relevant information, it can even be used as training data to define models and improve existing technologies. Training data must be of high quality, because it is essential to achieve the business objectives (Sarkis, 2023).

In conclusion, it can be stated that with the exponential increase in data, especially on the Internet, and the constant evolution of information technologies, the creation of new processes which allow for the proper analysis of big data is becoming increasingly necessary. Therefore, it is essential to create new processes to be able to treat big data and perform analysis automatization.

Acknowledgments

My heartfelt thanks to the Escuela Politécnica Nacional of Ecuador (EPN) for their unwavering support throughout this research journey.

Conflicts of Interest

The author has declared that no competing interests exist.

REFERENCES

Abramowicz, W. & Tolksdorf, R. (2010). Business information systems. *13th International Conference*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-12814-1

Aggarwal, C. C., & Zhai, C. (Eds.). (2012). *Mining text data*. Springer New York.

Almeida, I. (2023). *Introduction to Large Language Models for business leaders: Responsible AI strategy beyond fear and hype*. Now Next Later AI.

Amerland, D. (2013). *Google Semantic Search: Search Engine Optimization (SEO) Techniques that get your company more traffic, increase brand impact, and amplify your online presence*. Pearson Education.

Balusamy, B., Abirami, R. N., Kadry, S., & Gandomi, A. H. (2021). *Big Data: Concepts, Technology, and Architecture*. John Wiley & Sons.

Bao, Z., Borovica-Gajic, R., Qiu, R., Choudhury, F., & Yang, Z. (Eds.). (2023). Databases theory and applications. *34th Australasian Database Conference (ADC 2023)*. Springer Nature Switzerland.

Berry, M. W., & Kogan, J. (Eds.). (2010). *Text Mining: Applications and theory*. John Wiley & Sons.

Bobadilla, J. (2021). *Machine Learning y Deep Learning: Usando Python, Scikit y Keras*. Ediciones de la U.

Bustamante, N., & Guillén, S. (2020). *Big Data y Mass Media*. Aula Magna Proyecto clave McGraw Hill.

Campesato, O. (2023). *Transformer, BERT, and GPT3: Including ChatGPT and Prompt Engineering.* Mercury Learning and Information.

Cevallos, F. (2024, April 9). *GitHub dataset for digital news classification and punctuation using Machine Learning and Text Mining techniques*. Github, Inc. Retrieved from https://github.com/fcevallosepn/news

Chen, J., Huynh, V.-N., Tang, X., & Wu, J. (Eds.). (2023). Knowledge and systems science. *22nd International Symposium*. Springer Nature Singapore.

De Ville, B. (2001). *Microsoft data mining: Integrated business intelligence for e-commerce and knowledge management*. Digital Press.

Gils, B. (2023). *Data in context: Models as enablers for managing and using data*. Springer Nature Switzerland.

Gorelik, A. (2019). *The Enterprise Big Data lake: Delivering the promise of Big Data and data science*. O'Reilly Media.

Hildebrandt, M., & Gutwirth, S. (2008). *Profiling the European citizen: Cross-disciplinary*. Springer Netherlands.

Johri, P., Verma, J. K., & Paul, S. (Eds.). (2020). *Applications of Machine Learning (Algorithms for Intelligent Systems)*. Springer Nature Singapore.

Kannan, R., Rasool, R. U., Jin, H., & Balasundaram, S. R. (Eds.). (2016). *Managing and processing Big Data in cloud computing*. IGI Global. https://doi.org/10.4018/978-1-4666-9767-6

Koul, N., (2023). *Prompt engineering for Large Language Models*. Nimrita Koul.

Kumar, S. (2020). Can webometrics predict the academic rankings of institutes? *The Journal of Prediction Markets, 14*(2), 61-76. https://doi.org/10.5750/jpm.v14i2.1816

Nisbet, R., Miner, G., & Yale, K. (2017). *Handbook of statistical analysis and data mining applications*. Elsevier Science.

Ortega, J. M. (2022). *Big data, machine learning y data science en python*. RA-MA S.A. Editorial y Publicaciones.

Pasupuleti, P., & Purra, B. S. (2015). *Data Lake Development with Big Data*. Packt Publishing.

Rahman El Sheikh, A. A., & Alnoukari, M. (Eds.). (2012). *Business Intelligence and Agile Methodologies for Knowledge-Based Organizations: Cross-Disciplinary Applications*. IGI Global. https://doi.org/10.4018/978-1-61350-050-7

Rajaguru, H., & Prabhakar, S. K. (2017). *KNN classifier and K-Means clustering for robust classification of epilepsy from EEG signals. A detailed analysis*. Anchor Academic Publishing.

Ribeiro, J. A. (2019). *Big Data for executives and market professionals - Second edition*. Amazon Digital.

Rúa Pérez, J. (2009). *Tecnologìa, innovación y empresa*. Lulu Press, Incorporated.

Sánchez Trujillo, M., & Pérez Hernández, J. A. (2021). Metodología CRISP-DM en la gestión de proyecto de Data Mining. Caso enfermedades dermatológicas. *International Conference on Project Management*. EAN Universidad.

Sarkis, A. (2023). *Training Data for Machine Learning*. O'Reilly Media.

Suganthi, K., Karthik, R., Rajesh, G., & Ching, P. H. C. (Eds.). (2021). *Machine Learning and Deep Learning techniques in wireless and Mobile Networking Systems*. CRC Press.

Wang, L., Licheng, J., Shi, G., Li, X., & Liu, J. (Ed.). (2006). Fuzzy systems and knowledge discovery. *Third International Conference*. Springer Berlin Heidelberg.

Zong, C., Xia, R., & Zhang, J. (2021). *Text Data Mining*. Springer Nature Singapore.