Data Triage in News Articles Classification

Prateek Wadhwani
North Carolina State University
pwadhwa@ncsu.edu

Rohan Jigarbhai Shah
North Carolina State University
rshah29@ncsu.edu

Siddhant Gupta
North Carolina State University
sgupta45@ncsu.edu

Kaivan Ketan Shah
North Carolina State University
kshah23@ncsu.edu

1 Background

Data is one of a company's most valuable resources. The extraction of meaningful information from data to enhance business analytics has grown increasingly time intensive as data collection activities have increased across all sectors. Data triage is a technique for reducing the size of a dataset in order to achieve equivalent results with less computational time and resources. The goal of this project is to perform Data Triage on News Articles in order to minimize the original dataset while maintaining similar level of accuracy.

1.1 Literature Survey

In this paper [1], the author used the Tf-Idf method of duplicate detection to decrease the data before passing it through a Naive Bayes classifier. The technique works well for their dataset as they are working with Bugs detection for software products in which mainly the consumer use the same set of words to inform about the product bugs. But when it comes to News Articles, we cannot only depend on a particular set of words to find the duplicates. The same article about the same person can provide different news altogether.

Near-duplicate documents are particularly common in news media corpora [2]. Editors often update wirefeed articles to address space constraints in print editions or to add local context; journalists often lightly modify previous articles with new information or minor corrections. Near-duplicate documents have potentially significant costs, including bloating corpora with redundant information (biasing techniques built upon such corpora) and requiring additional human and computational analytic resources for marginal benefit. Filtering near-duplicates out of a collection is thus important, and is particularly challenging in applications that require them to be filtered out in real-time with high precision.

In order to classify the data and detect similarity among the different news articles, we had to understand the structure of the language, the effects of the various models on it and the suitable structures which could help enhance the similarity detection.[3] The explanation of models like GloVE, TF-IDF, Jaccard and Doc2Vec for similarity detection and SVM, Logistic Regression and LSTM for classification in the survey by Kowsari et. al. was necessary for us to understand the use of the concepts being applied in these models. Sample examples on small text sentences showcased concepts like vectorization and how they could be used in similarity detection in models like GloVe. The survey about the text classification algorithms helped comprehending the use of various machine learning algorithms. Post that, we were able to apply these algorithms on our dataset, and this helped us narrow down the applications of each one of them which would be relevant in our use-case.

2 Proposed Method

2.1 Approach

Our approach for data triage:

- The original data is divided into training and testing sets. We initially train our models using the unaltered training dataset and perform classification of the test dataset.
- Document similarity algorithms are implemented on the training data set to produce our training subsets. For each similarity algorithm, we set different thresholds based on our knowledge of the model and a series of experiments of the model.

Method	Threshold
Jaccard	60%
Tf-Idf + Cosine	70%
GloVe + Cosine	95%
Doc2Vec + Cosine	80%

- Post dataset reduction, we follow a similar pipelines of operations by training our model on the data subset and performing classification of the test data.

The following methods have been used for data classification:

- Logisitic Regression: It is a process of modeling the probability of a discrete outcome given an input variable. Logistic regression is a useful analysis method for classification problems, where you are trying to determine if a new sample fits best into a category. Logistic Regression was used with both TF-IDF vectorization method. The text is tokenized and converted into its corresponding vector before being fed into the logistic regression model which later predicts the class of the test news articles.
- Naive Bayes: A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem and the information of the probability of the word occurrences is given by our vectorization technique. The text is tokenized and converted into its corresponding vector before being fed into the bayesian classifier model.
- SVM (Support Vector Machine): The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N — the number of features) that distinctly classifies the data points. Our objective is to find a plane that will help classify our news articles into different categories. SVM was used with TF-IDF vectorization which transformed the data into a vector format which could be used for classification.
- Long Short-Term Memory: A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature. Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria. The neural networks model is used for tokenization, text vectorization and as a classification model which runs for ten iterations learning from every iteration giving an optimal accuracy.

The following methods have been used for text vectorization:

- TF-IDF: Term Frequency-Inverse Document Frequency (TF-IDF) is a common method to evaluate how important a single word is to a corpus. The TF-IDF method can assess the importance of each word in the corpus by taking into account the number of its occurrences. It can be obtained by multiplying the number of times a word appears in a document, and the inverse document frequency of the word across a set of documents. TF-IDF will output the vectors that are required for classification and similarity measures that are used in the project.

The following methods have been used for detecting document similarity:

- Jaccard Similarity: Jaccard Similarity coefficient, also sometimes referred to as Jaccard Index, is Intersection over Union. For the project, we have used the Trigram approach where the entire article is divided into sets of 3 words, known as trigrams, and then the trigrams of the articles are compared for obtaining similarity.
- Cosine Similarity: Cosine Similarity is a measurement that quantifies the similarity between two or more vectors. The cosine similarity is the cosine of the angle between vectors [9]. Suppose the angle between the two vectors is 90 degrees, the cosine similarity will have a value of 0; this means that the two vectors are perpendicular to each other which means they have no correlation between them [10]. In our implementation, we have used Tf-Idf in conjunction with cosine similarity to identify similarity between the news articles to form the basis for subset creation.
- GloVe Embeddings: GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space [11]. In our application, we have used Glove Embedding alongside cosine similarity to identify similarity between the news articles to form the basis for subset creation.
- Doc2Vec Embeddings: Doc2Vec is a powerful NLP tool to represent documents as a vector. Doc2Vec model is based on the Word2Vec model. While Word2Vec computes a feature vector for every word in the corpus. Doc2Vec computes a feature vector for every document in the corpus. Doc2Vec is also used with cosine similarity to create our training data subsets.

2.2 Rationale

We examine whether two articles are sufficiently similar using different Document Similarity techniques, and whether eliminating one of them would have minimal effect on the performance of the classification models used. We can decrease the original dataset by comparing all of the News Articles to one another and deleting similar or roughly duplicate News Articles, resulting in the classification models requiring substantially lesser computational resources and time and provide nearly same accuracy as implemented on the original dataset.

3 Plan and Experiment

We have used open source news articles datasets and one proprietary dataset from aylien. We collated articles for the following categories: Finance, Entertainment, Politics, Technology and Travel. The source data was originally available as .json files. For the ease of processing, we pre-processed and stored the data in .csv format.

3.1 Data Sourcing

The dataset format for News Articles from webhose.io [1], [2], [3], [4] is as below:
Features: organization (string), uuid (string), author (string), url (string), ord_in_thread (string), title (string), locations (string), highlightText (string), language (string), text (string), published (date), crawled (date), highlightTitle (string)

The dataset format for Political News Articles from POLUSA [6] is as below: Features: id (integer), date_publish (date), outlet (string), headline (string), lead (string), body (string), body (string), authors (string), domain (string), url (string), political_leaning (string)

The dataset format for Financial News Articles from Aylien [5] is as below:
Features: author (string), body (string), categories (string), character_count (integer), clusters, entities, hashtags, keywords (string), language (string), links (string), media (string), paragraph_count (integer), published_at (date), sentences_count (integer), sentiment (string), social_shares_count (integer), source (string), summary (string), title (string), words_count (integer)

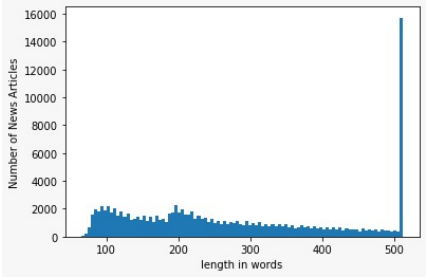
3.2 Data Description

Features: Class (indicating the news category), Title (news article title), News (news article)
After dropping all the records with missing values, we obtained the clean dataset. The below table shows the number of records in each category:

Category	Count
Politics	865190
Travel	49469
Tech	87157
Sports	156899
Finance	40126
Entertainment	50282

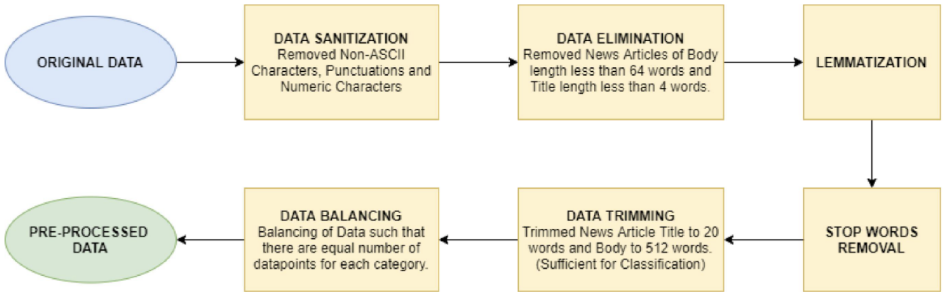
As visible in the above table, the record counts for Politics News Articles is much higher than the remaining categories. Thus, data balancing across the categories was required.

The below graph represents the length of the news articles in the dataset. As visible in the graph, there are a lot of news articles with length of more than 512 words.Thus, for better computation on such a huge dataset, we trimmed the news articles body to a maximum of 512 words. Doing this would not affect the results of the classification substantially as 512 words would be enough for any news articles to be classified into one of the six categories in our dataset.



3.3 Data Preprocessing:

From the raw data, only the columns ‘Title’, ‘News’ and ‘Category’ were retained since only these columns are relevant for the Data Triage and Classification. On the dataset, we performed Data Sanitization in which the Non-ASSCII characters, Punctuations and Numeric characters were removed. Then all the News Articles with Body length less than 64 words or Title length less than 4 words were removed as shorter articles did not add significant value to the classification. Then, by using the NLTK library, Lemmatization was performed and Stop Words were removed from the Title and Body of each article. As the main goal of our project is to get good classification results on the Triaged dataset, we trimmed each article body to maximum of 512 words and each title to maximum of 20 words. To obtain the pre-processed data set on which we can perform classification, we balanced the data such that there were equal number of datapoints for each of the six categories of News we have. This is done so that the model can be trained without any bias towards a particular category which would have resulted from an unbalanced dataset.



3.4 Data Summarization and Visualization:

The below figure is a small representation of the preprocessed dataset. The text obtained after preprocessing is clean of all the Non-ASCII characters, punctuations, numeric characters and stop words.

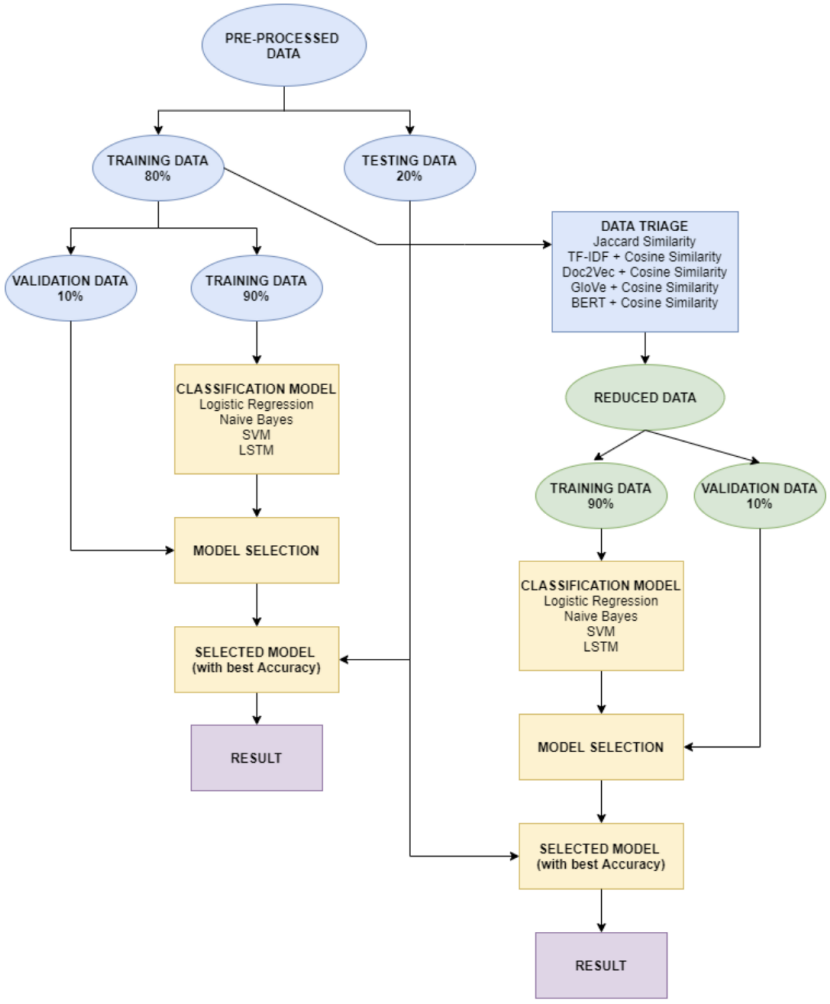
Class	Title	News
Politics	hammond hit armed robbery	police arrested three suspect connection two h...
Politics	rocket howard say played torn mcl	rocket howard say played torn mcl center say d...
Politics	trump condemns egregious display hatred bigotr...	president trump saturday condemned egregious d...
Tech	raven special teamer brynden trawick seeing ti...	published baltimore sun sport today starting s...
Tech	review question ipart ruling holroyd holroyd c...	source holroyd city council october medium rel...

3.5 Hypothesis

The hypothesis for our experiment is to obtain similar results with a smaller training dataset which forms the basis for Data Triaging. It states that a smaller training dataset may provide equivalent if not more information to the models to help them correctly classify the News Articles into our given categories. Training the models on a subset of the data is expected to decrease the computational operations which may introduce a trade-off with a reduction in accuracy. The main focus of this project is to identify the best method for data reduction that will retain the most information provided by our training dataset. Additionally, we want to determine the decline in the accuracy if any of our classification models when training with a smaller dataset and decide if the trade-off between accuracy and computational time is warranted.

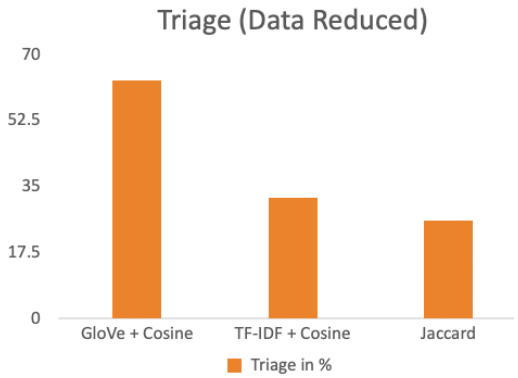
3.6 Experiment Setup:

The below flowchart demonstrates the experimental setup used to perform Data Triage methods and Classification on the preprocessed dataset. First, the preprocessed data is split into training and testing data in the ratio 80:20. Classification algorithms described above are implemented and accuracy is obtianed using the testing dataset. Then, the Document Similarity models are executed on the same training data and a reduced dataset (triaged dataset) is obtained. The same classification models are implemented on the new training data. The accuracy of the classification models are obtained using the same testing dataset. The results of each classification model before and after Data Triage are compared and the results are analyzed.

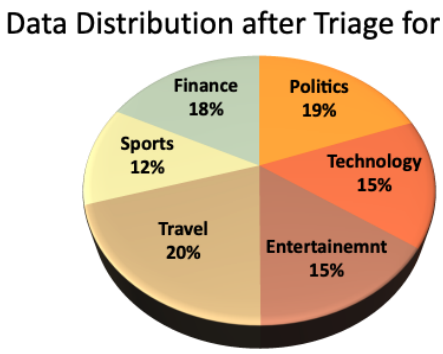


4 Results

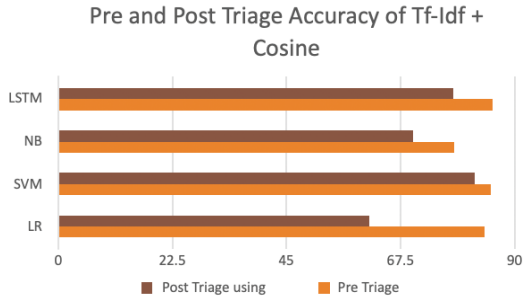
We implemented Term Frequency - Inverse Document Frequency (TF-IDF) word vectorization with Logistic Regression, Naive Bayes Classification, Support Vector Machine Classification and Recurrent Neural Networks using Long-Short Term Memory (LSTM).



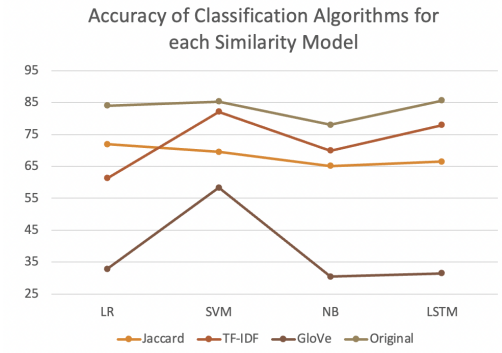
GloVe Embedding was found to be most effective in performing data triage.



All classification algorithms were implemented on our reduced dataset, obtained using Tf-Idf + Cosine Similarity Method.



Consequently, we implemented the classification algorithms for the reduced dataset obtained using all the similarity methods and obtained interesting results.



4.1 Critical Evaluation

At the start of the project, we assumed that the model which can best detect the contextual similarity between news articles would be the best technique for the Data Triage as it would remove the articles which would add no significance to the classification model's training process. So, we assumed that GloVe Embeddings would perform the best for our problem statement. However, in the results we obtained, we observed that Data Triage using GloVe Embeddings had the least testing accuracy for all classification models.

We address the decline in accuracy by considering the performance of our SVM model with the subset created by Tf-Idf and Consine Similarity Method. It has the highest accuracy and only drops seven percent compared to the accuracy obtained by our original dataset. Given that the above similairty method causes triaging of about 35%, considerably reducing the computational operations required for training our our model, the accuracy drop seems justifiable. However, this conclusion of a warranted trade-off is an open ended question and the answer to it depends on the use-case.

5 Conclusion

- We obtained a better grasp of the working of Document Similarity Algorithms such as Jaccard Similarity, GloVe Embeddings, TF-IDF, and Doc2Vec Model during the term project on Data Triage for News Articles Classification. We were able to identify the algorithms' flaws and how other algorithms would address them.
- GloVe was efficient in finding the contextual meaning (rather than the semantic meaning) between News Articles which gave it the upper hand in detecting Similar News Articles. However, a lot of articles were removed which decreased the classification accuracy post data triage.
- TF-IDF detects the ranking of words in different documents and using it in conjunction with the Cosine Similarity was able to retain most information of the training dataset and provide highest accuracy when classifying the articles in our test dataset.
- Bi-directional LSTM with GloVe Embeddings is an efficient classification model provided that there is sufficient training data.
- Due to the time constraint and the lack of computational resources required to implement Doc2Vec Model, we could not successfully obtain the results on the entire dataset. The code for the same is available on our Github repository. However, we tested it on a small portion of the dataset and found it to be an effective model for detecting document similarity. As a result, we believe it can be a useful tool for performing data triage on large datasets if appropriate computational resources and time are available.

6 References

1. Rajeshwari M R & Dr. Kavitha K S (2017) Reduction of Duplicate Bugs and Classification of Bugs using a Text Mining and Contingency Approach. International Journal of Applied Engineering Research
2. Simon Rodier & Dave Carter (2020) Online Near-Duplicate Detection of News Articles Digital Technologies, National Research Council Canada
3. Kowsari, Kamran, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes, and Donald Brown. 2019. "Text Classification Algorithms: A Survey" Information 10, no. 4: 150. https://doi.org/10.3390/info10040150
4. Webhose.io (2015) https://webhose.io/free-datasets/technology-news-articles/
5. Webhose.io (2015) https://webhose.io/free-datasets/entertainment-news-articles/
6. Webhose.io (2015) https://webhose.io/free-datasets/travel-news-articles/
7. Webhose.io (2015) https://webz.io/free-datasets/sports-news-articles/
8. Aylien News API Financial Crimes Dataset https://aylien.com/blog/free-dataset-downloads-natural-disasters-financial-crimes-nasdaq-100
9. Lukas Gebhard, Felix Hamborg. 2020. The POLUSA Dataset: 0.9M Political News Articles Balanced by Time and Outlet Popularity
10. Susan Li (2019) Multi-Class Text Classification with LSTM, towardsdatascience.com
11. Gunter Rohrich (2020) Find Text Similarities with your own Machine Learning Algorithm, towardsdatascience.com
12. Jair Neto (2021) Best NLP Algorithms to get Document Similarity, medium.com
13. Sindhu Seelam (2021) Machine Learning Fundamentals: Cosine Similarity and Cosine Distance, medium.com
14. Jeffrey Pennington, Richard Socher, Christopher D. Manning (2014) GloVe: Global Vectors for Word Representation, nlp.stanford.edu

Github Repository Link: https://github.ncsu.edu/sgupta45/engr-ALDA-fall2021-P09