AI-Written vs Human-Written: Classification of Scientific Abstracts

Francesco Galardi

Contents

1 Introduction

With the rapid advancement of artificial intelligence and the emergence of generative language models such as ChatGPT—capable of producing increasingly coherent, articulate, and realistic texts—it has become progressively more challenging to distinguish between content written by humans and that generated by machines.

This project aims to tackle this issue by focusing on the classification of abstracts from medical-scientific articles, in order to determine whether they were written by human authors or generated by ChatGPT.

The proposed approach relies on text mining and classification techniques: various classification models, combined with different text representation strategies, will be tested and compared to identify the most accurate and effective solution.

2 Dataset

The dataset consists of 28,662 observations, evenly distributed between class 0 (human-written) and class 1 (ai-written), making it a perfectly balanced dataset. The features included are: title, abstract, and label.

Preliminary analysis reveals that each title appears exactly twice, once paired with an abstract labeled as 0 and once with an abstract labeled as 1. This indicates that for every title, both human-written and AI-generated versions of the abstract are present.

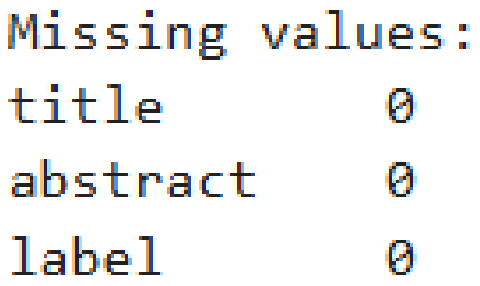

Figure 1: No missing values in the dataset

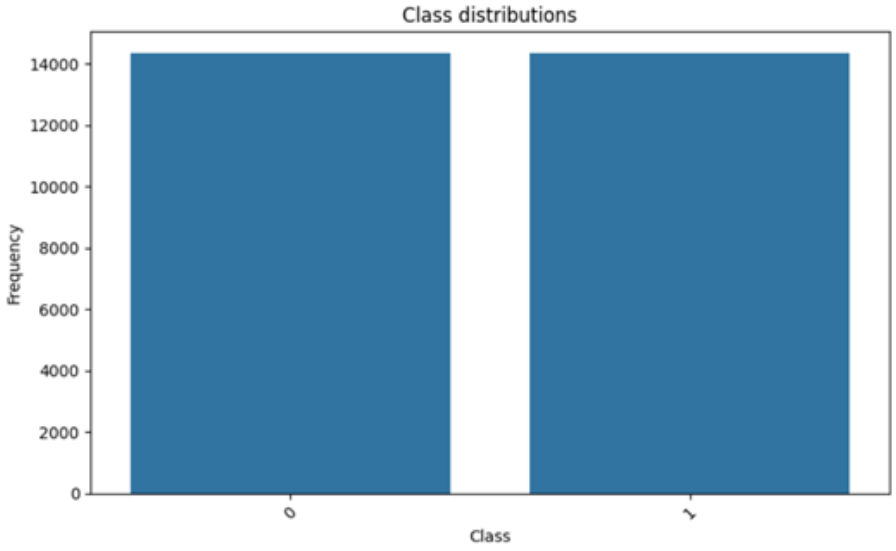

Figure 2: Class distribution

```
Numero di titoli diversi nel dataset: 14331
Numero di titoli unici per ogni classe:
label
0    14331
1    14331
```

Figure 3: of different titles per class

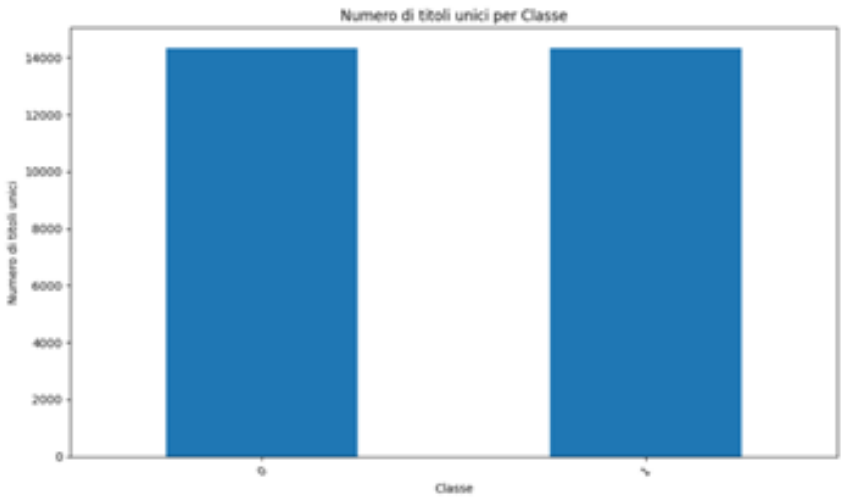

Figure 4:

A final look at the word cloud of the most frequent terms in the dataset provides further confirmation of the thematic context and the topics covered in both the abstracts and their corresponding titles.

This visualization allows for a quick grasp of the most recurrent terms, offering an immediate insight into the medical and scientific nature of the content being analyzed.

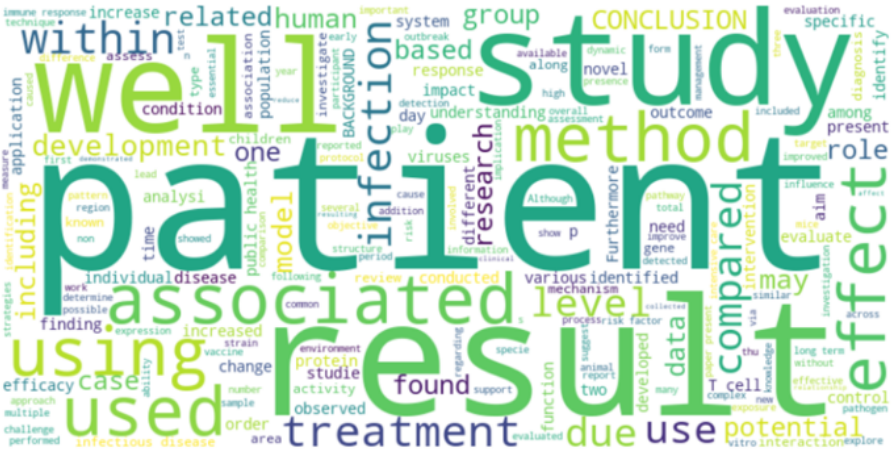

Figure 5: WordCloud of most frequent words

3 Data Preprocessing

The dataset described above has also been used for the same purpose in the related work referenced at the end of this documentation. However, in this project, the focus is placed on classifying abstracts solely based on their lexical and syntactic characteristics, that is, their stylistic form.

That said, the original structure of the dataset may not be entirely suitable for this objective, for the following reasons:

- A model might learn to classify an abstract based on the semantic association between the abstract and its title, rather than its linguistic form.
- Since each title is associated with two abstracts—one labeled 0 and one labeled 1—the model could end up distinguishing between abstracts based on their stylistic difference within a certain topic given by the title, not their general writing style.

To mitigate these issues, for each title, only one abstract was randomly selected, choosing either the version with label 0 or label 1. After this selection, the title feature was removed entirely.

The resulting dataset, containing half of the original observations, remains balanced, as random selection from two equally sized sets does not significantly alter the overall distribution of the two classes. This is visually confirmed in the figure below. An 80-20 split was then applied to divide the dataset into a training set and a test set for the modeling phase.

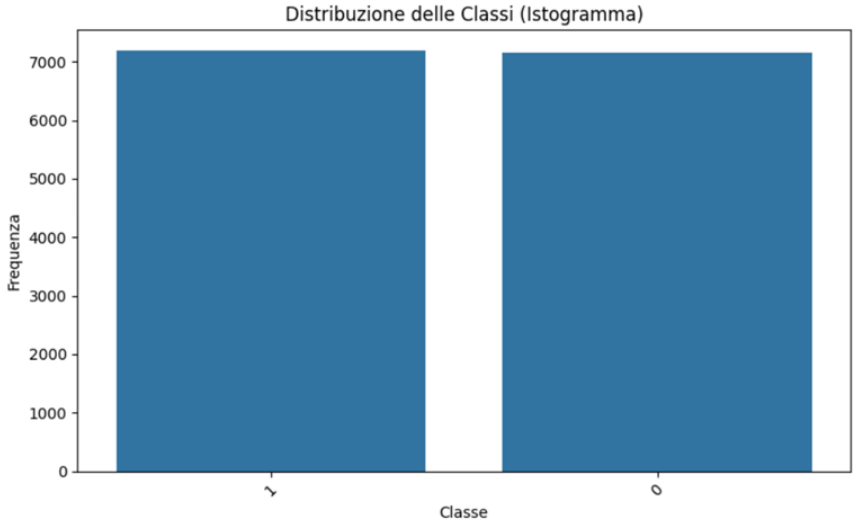

Figure 6: Class distribution after preprocessing

4 Text Mining Classification

Regarding text representation, two main approaches were considered: TF-IDF and Word2Vec.

More advanced models focused on syntactic understanding, such as BERT, were intentionally excluded for specific reasons.

As explained in the related work mentioned at the end, the dataset was built by collecting scientific abstracts from a medical article database created in response to the COVID-19 pandemic. For each article title, an alternative abstract was generated by AI—presumably within one or more closely timed ChatGPT sessions.

The risk of using models like BERT lies in their ability to overfit to the specific syntactic patterns used during that particular generation session, thus reducing their generalization capability when classifying abstracts produced in different contexts or at different times/sessions.

To address this, the focus shifted toward more temporally stable linguistic traits, such as vocabulary and general structural patterns, which tend to persist across generations by the AI. For this reason, TF-IDF and Word2Vec were chosen as the main text representation techniques.

The classification models used include:

- Logistic Regression
- Naive Bayes (TF-IDF only)
- Decision Tree
- Support Vector Classifier (Linear)
- Random Forest
- XGBoost

Each model was tested with both representation methods (except Naive Bayes) using nested cross-validation:

- 5-fold inner cross-validation for hyperparameter tuning
- 10-fold outer cross-validation for performance evaluation

Given the binary and balanced nature of this classification task, accuracy was chosen as the primary evaluation metric. Each pipeline tested includes the following stages:

- 1. Tokenization
- 2. Stopword removal, including standard English stopwords, punctuation symbols, mathematical symbols and years up to the current
- 3. Text vectorization (TF-IDF or Word2Vec)
- 4. Classification model

Stemming and lemmatization were deliberately excluded, as these techniques could distort class-specific lexical cues (e.g., verb tenses or specific word forms used more frequently in one class).

For the same reason, only minimal text cleaning was applied—limited to removing non-informative elements.

Finally, we report the average accuracy, standard deviation, and optimal hyperparameters identified via GridSearch within the nested cross-validation process.

- Tf-idf+logistic regression:

```
Accuracy fold-by-fold: [0.95466434 0.95466434 0.95989538 0.96338274 0.95636998 0.95549738
 0.96335079 0.94589878 0.96858639 0.95724258]
Media accuracy: 0.9580
Deviazione standard: 0.0059
{'clf__C': 1}
```

Figure 7:

- Tf-idf+nayve bayes:

```
Accuracy fold-by-fold: [0.88666085 0.85265911 0.87183958 0.87619878 0.86823735 0.86910995
 0.88394415 0.87870855 0.89179756 0.86038394]
Media accuracy: 0.8740
Deviazione standard: 0.0114
{'clf__alpha': 0.1}
```

Figure 8: Enter Caption

- Tf-idf+decision tree:

```
Accuracy fold-by-fold: [0.87968614 0.88404534 0.88404534 0.90671316 0.87958115 0.89092496
 0.88917976 0.88045375 0.89703316 0.88394415]
Media accuracy: 0.8876
Deviazione standard: 0.0083
{'clf  max depth': 10}
```

Figure 9:

- Tf-idf+svc:

```
Accuracy fold-by-fold: [0.95640802 0.95640802 0.9625109  0.96163906 0.96335079 0.95375218
 0.96596859 0.94851658 0.97033159 0.96073298]
Media accuracy: 0.9600
Deviazione standard: 0.0060
{'clf__C': 1}
```

Figure 10:

- Tf-idf+random forest:

```
Accuracy fold-by-fold: [0.9537925  0.95030514 0.93374019 0.95466434 0.94415358 0.95113438
 0.95811518 0.94589878 0.96247818 0.94851658]
Media accuracy: 0.9503
Deviazione standard: 0.0076
{'clf  max depth': 20, 'clf  n estimators': 200}
```

Figure 11:

- Tf-idf+xgboost:

```
Accuracy fold-by-fold: [0.9625109  0.96774194 0.95989538 0.97297297 0.96945899 0.96422339
 0.96684119 0.95636998 0.97382199 0.96596859]
Media accuracy: 0.9660
Deviazione standard: 0.0052
{'clf__max_depth': 3, 'clf__n_estimators': 200}
```

Figure 12:

- W2Vec+logistic regression:

```
Accuracy fold-by-fold: [0.95292066 0.94768963 0.95902354 0.95117698 0.94240838 0.95636998
 0.94938918 0.94153578 0.95811518 0.95200698]
Media accuracy: 0.9511
Deviazione standard: 0.0057
{'clf__C': 10}
```

Figure 13:

- W2Vec+decision tree:

```
Accuracy fold-by-fold: [0.83958152 0.81778553 0.83522232 0.82476024 0.84991274 0.81326353
 0.83769634 0.82984293 0.84991274 0.84904014]
Media accuracy: 0.8347
Deviazione standard: 0.0125
{'clf__max_depth': 10}
```

Figure 14:

- W2Vec+svc:

```
Accuracy fold-by-fold: [0.96512642 0.95640802 0.9625109  0.95989538 0.95200698 0.96247818
 0.95200698 0.95462478 0.96247818 0.95200698]
Media accuracy: 0.9580
Deviazione standard: 0.0049
{'clf  C': 100}
```

Figure 15:

- W2Vec+random forest:

```
Accuracy fold-by-fold: [0.90932868 0.89712293 0.9163034  0.91281604 0.91361257 0.90750436
 0.90837696 0.91012216 0.90924956 0.90139616]
Media accuracy: 0.9086
Deviazione standard: 0.0054
{'clf__max_depth': 20, 'clf__n_estimators': 200}
```

Figure 16:

- W2Vec+xgboost:

```
Accuracy fold-by-fold: [0.95117698 0.93548387 0.95466434 0.95292066 0.94764398 0.94851658
 0.95811518 0.95287958 0.95200698 0.94764398]
Media accuracy: 0.9501
Deviazione standard: 0.0058
{'clf__max_depth': 3, 'clf__n_estimators': 200}
```

Figure 17:

Since the two text representation techniques used—TF-IDF and Word2Vec—are conceptually different, we decided to select the top three performing models, optimized with their best parameters founded in previous step, for each. The models achieving the highest accuracy were Logistic Regression, SVC, and XGBoost, for both TF-IDF and Word2Vec representations.

Based on this selection, pairwise comparisons between models were conducted using statistical tests on their fold-by-fold accuracy scores. Given the overall high accuracy of these models, statistically significant differences would suggest distinct performance distributions—indicating different error localization. This supports the hypothesis that combining statistically diverse models in an ensemble could leverage their complementarity, potentially improving both overall accuracy and generalization to unseen or unlabeled abstracts.

The comparison methodology involved computing the fold-by-fold accuracy differences for each model pair. If the resulting residuals followed a normal distribution (verified via the Shapiro-Wilk test), a paired t-test was applied; otherwise, the Wilcoxon signed-rank test was used.

To ensure a sufficient number of samples, each model (already optimized through nested cross-validation as said before) was evaluated by repeating the cross-validation three times.

	vett1logreg	vett2logreg	vett1svc	vett2svc	vett1xgb	vett2xgb
vett1logreg						
vett2logreg	gaussian					
vett1svc	gaussian	gaussian				
vett2svc	gaussian	gaussian	gaussian			
vett1xgb	gaussian	gaussian	gaussian	gaussian		
vett2xgb	gaussian	gaussian	gaussian	gaussian	gaussian	

	vett1logreg	vett2logreg	vett1svc	vett2svc	vett1xgb	vett2xgb
vett1logreg						
vett2logreg	Different					
vett1svc	Different	Different				
vett2svc	Equal	Different	Equal			
vett1xgb	Different	Different	Different	Different		
vett2xgb	Different	Different	Different	Different	Different	

Figure 18: First matrix: M1[i,j] = result of the shapiro-wilk test of residuals of th accuracy of the two models. Second matrix: M2[i,j] = result of statystic test (t-test of Wilcoxon) between two models. Note that vett1... stand for the vector of accuracies of a model with tf-idf representation while vett2 the same thing but with word2Vec representation

Based on statistical test outcomes, two ensemble strategies were proposed:

- Exclude TF-IDF + SVC and TF-IDF + Logistic Regression and maintain all the rest
- Exclude Word2Vec + SVC and maintain all the rest

Both ensemble variants will be evaluated using standard cross-validation. Each model that will participate in the ensemble will be used with the optimal hyperparameters identified during the previous grid search. Finally, the two ensembles will be statistically compared using their fold-wise accuracy scores. If a significant difference is found, the better-performing ensemble will be selected.

```
Accuracy fold-by-fold: [0.97384481 0.9668701  0.98169137 0.97646033 0.96858639 0.96945899
 0.97382199 0.96684119 0.97294939 0.97294939]
Media accuracy: 0.9723
Deviazione standard: 0.0044
Accuracy fold-by-fold: [0.97907585 0.97297297 0.97994769 0.97646033 0.97120419 0.96771379
 0.97643979 0.96335079 0.98080279 0.97731239]
Media accuracy: 0.9745
Deviazione standard: 0.0054
```

Figure 19:

	vett_ensemble1	vett_ensemble2
vett_ensemble1		
vett_ensemble2	gaussian	

	vett_ensemble1	vett_ensemble2
vett_ensemble1		
vett_ensemble2	Different	

Figure 20:

5 Conclusion

By evaluating the images above, we observe that the two models are statistically different. Therefore, we choose the one that achieved the best results in terms of accuracy, even if only slightly better—namely, the second ensemble method.

To summarize, the final chosen method is an ensemble method that bases its prediction on a voting system composed of the following models:

- TF-IDF + Logistic Regression
- TF-IDF + SVC
- TF-IDF + XGBoost
- Word2Vec + Logistic Regression
- Word2Vec + XGBoost

The voting system is of the soft type, meaning that each classifier outputs the probability that the new abstract belongs to class 0 or class 1. For each class, the average of the probabilities from all classifiers is computed. The class with the highest average probability is then selected as the predicted class.

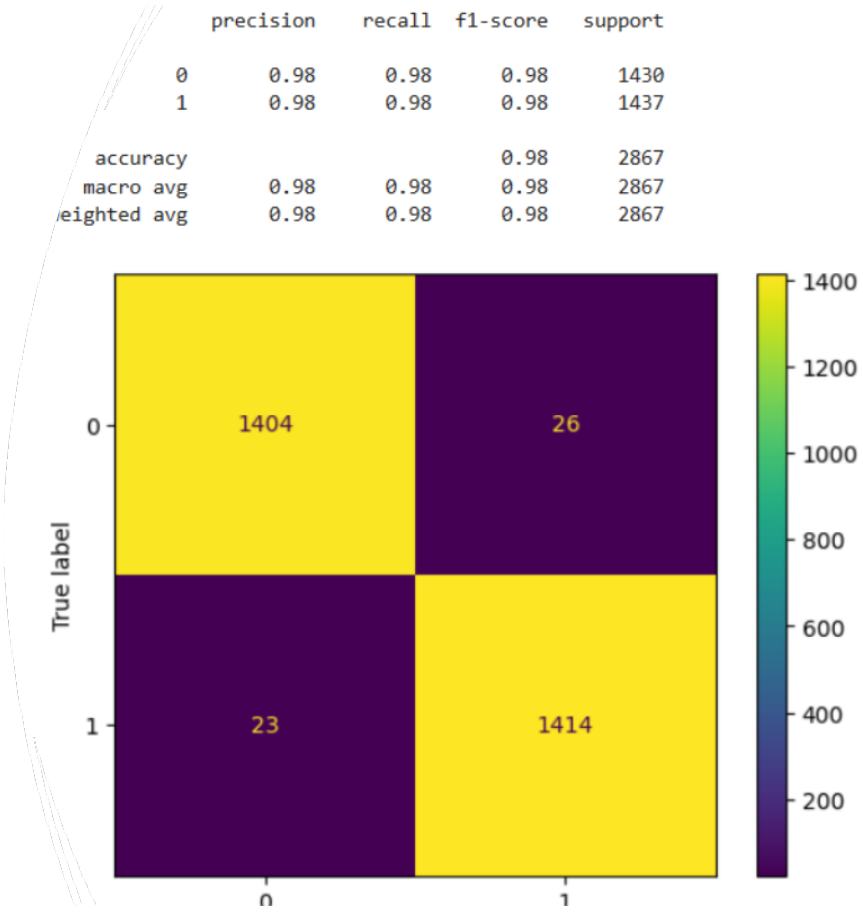

Figure 21: Confusion matrix of the prediction of the final ensemble on the test set fitted on the training set

6 Interface

The image below shows a graphical interface where the final ensemble can be tested with unseen abstract

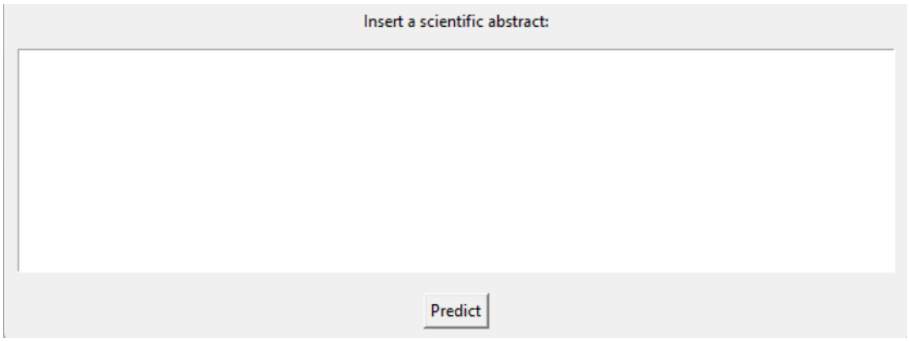

Figure 22: GUI

7 References

1. Theocharopoulos, P. C., Anagnostou, P., Tsoukala, A., Georgakopoulos, S. V., Tasoulis, S. K., & Plagianakos, V. P. (2023, April 12). *Detection of fake generated scientific abstracts*. arXiv:2304.05961