RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

XG BOOST MODEL - BASED ALPHA, SIGNAL PREDICTION USING MICROBOLOGGING DATA FROM SOCIAL MEDIA

M. Amareshwar[1], K. Shivani[2], B.V. Krishna Sai[2], U. Nagaraj[2]

[1]Assist. Professor, [2]UG Scholar, [1,2]Department of Computer Science & Engineering (Data Science),

[1,2]Kommuri Pratap Reddy Institute of Technology, Ghatkesar, Hyderabad, Telangana.

ABSTRACT

Alpha signals, also referred to as excess returns, play a crucial role in assessing the performance of a financial asset compared to the overall market benchmark. The ability to predict alpha signals holds immense value accurately and promptly for investors and financial analysts, as it can greatly influence portfolio optimization and risk management decisions. However, traditional methods of alpha signal prediction, which heavily rely on historical financial data, have their limitations in capturing real-time market sentiments and changes. To overcome these limitations, researchers have started exploring alternative data sources, particularly social media data, to gain deeper insights into market sentiments and enhance alpha signal prediction. The integration of social media data into financial analysis presents an innovative approach to understanding investor sentiment, market perceptions, and collective behavior. Microblogging platforms such as Twitter and StockTwits serve as rich sources of real-time information, reflecting opinions and reactions to financial events as they happen. Leveraging such data for alpha signal prediction has the potential to complement and strengthen traditional financial analysis methods, leading to more precise and robust predictions. In light of this, the focus of this study is to utilize microblogging data from social media platforms to predict alpha signals in financial markets. The chosen approach employs the XGBoost model, a powerful machine learning algorithm renowned for its capability to handle complex and unstructured data with high dimensions. The model is trained using historical data and then tested on out-of-sample data to evaluate its predictive performance and accuracy. By harnessing the real-time and sentiment-rich information from social media, this proposed work aims to contribute to the advancement of alpha signal prediction methodologies and enhance decision-making processes in the financial domain.

Keywords: Boosting, Xgboost, NLP, TF-IDF, Tokenization.

1. INTRODUCTION

Alpha signal in the context of microblogging data from social media typically refers to a metric or value that is calculated to assess the significance or impact of a particular piece of content (such as a tweet or post) on a social media platform [1]. This signal is often used in social media analytics and sentiment analysis to measure the level of engagement, influence, or attention that a post has received within a specific community or network [2]. The specific calculation of the alpha signal can vary depending on the platform and the goals of the analysis, but it may consider factors such as:

— Engagement Metrics: Alpha signal may consider metrics like the number of likes, shares, comments, retweets, or favorites a post has received. These metrics can indicate how much other users are interacting with the content [3].

— Sentiment Analysis: The sentiment expressed in the comments or replies to a post can also be factored into the alpha signal. Positive or negative sentiment can provide insights into the overall reception of the content.

— Influence Metrics: Some algorithms for alpha signal calculation may consider the influence of the users who have interacted with the post. Users with a larger following or higher engagement rates themselves may contribute more to the signal [4].

— Time Decay: The recency of interactions may be taken into account, with more recent engagements being weighted more heavily than older ones.

— Network Structure: The position of the author within the social network (e.g., how many followers they have and their connections) may also be considered when calculating the alpha signal.

The alpha signal is useful for understanding which pieces of content are gaining traction and resonating with an audience, making it valuable for marketers [5], social media managers, and researchers. It can help identify trending topics, influential users, and the effectiveness of social media campaigns.

Research motivation for studying alpha signals in microblogging data from social media can be driven by several important factors and objectives. Alpha signals can help researchers gain a deeper understanding of how influence operates on social media platforms [6]. Investigating what types of content receive higher alpha signals and why certain users have a stronger impact can shed light on the dynamics of online influence. One key aspect of social media research is exploring why certain pieces of content go viral while others do not. Analyzing alpha signals can provide insights into the factors that contribute to content virality, helping marketers and content creators better tailor their strategies. By incorporating sentiment analysis into alpha signal calculations, researchers can study how the sentiment of interactions with content impacts its overall influence [7]. This can be particularly useful for tracking public opinion and understanding how it evolves over time.

Alpha signals can be used to identify key opinion leaders or influential users within specific niches or communities. Recognizing who holds the most sway within a particular network can be valuable for targeted marketing and outreach efforts. Businesses and organizations often invest heavily in social media marketing campaigns. Analyzing alpha signals can help assess the success of these campaigns by quantifying their impact and reach. Monitoring alpha signals can help identify emerging trends and breaking news topics in real-time. This can be beneficial for news organizations, government agencies, and businesses looking to stay informed and respond quickly to current events [8].

Research on alpha signals can reveal user engagement and interaction patterns. Understanding how users engage with content and with each other on social media can inform platform design, content recommendation algorithms, and community management strategies. Investigating how social media platforms calculate alpha signals can uncover potential biases in the way content is promoted or demoted. This research can contribute to discussions about algorithmic fairness and transparency [9].

Alpha signals can also be linked to social behavior and psychology. Research in this area can provide insights into why users engage with certain content, the emotional responses it elicits, and the role of social validation in online communities. Understanding the impact of influential content on social media is relevant to policy discussions and potential regulation. Research on alpha signals can inform policymakers about the effects of content on public discourse and behavior.

In the ever-evolving landscape of social media, understanding and predicting the factors that contribute to the emergence of alpha signals for specific content is crucial for marketers, content creators, and platform developers. Alpha signals, encompassing engagement metrics, sentiment analysis, and influence indicators, serve as valuable indicators of content success and audience impact [10]. However, there is a pressing need to develop robust predictive models that can anticipate the generation of alpha signals in real-time or shortly after content publication.

This research problem aims to address the following key challenges:

— **Dynamic Nature of social media:** Social media platforms are characterized by rapidly changing trends and user behaviors. Developing predictive models that can adapt to these dynamics and provide timely predictions of alpha signals is a complex problem.

— **Multi-factorial Nature of Alpha Signals:** Alpha signals are influenced by a multitude of factors, including content quality, user engagement, network structure, and sentiment. Building predictive models that accurately capture the interplay of these factors and their evolving importance is a challenging task.

— **Data Volume and Noise:** Social media platforms generate massive volumes of data, and not all interactions contribute equally to alpha signals. Filtering out noise and identifying meaningful patterns within the data is essential for accurate predictions.

— **Privacy and Ethical Considerations:** Handling social media data for predictive purposes raises privacy and ethical concerns. Researchers must navigate the ethical implications of data collection and usage, ensuring compliance with regulations and user consent.

— **Evaluation Metrics:** Defining appropriate evaluation metrics for alpha signal prediction models is crucial. Researchers need to establish metrics that align with practical use cases, such as marketing campaign optimization or content recommendation.

The overarching goal of this research is to develop predictive models that can forecast the likelihood of content receiving high alpha signals on social media platforms. Solving this problem has the potential to revolutionize content strategies, enabling content creators and marketers to tailor their efforts for maximum impact and engagement. Additionally, it can contribute to our understanding of the underlying dynamics of social media networks and user behavior in the digital age.

2. LITERATURE SURVEY

Simay, A. E., et al. (2023) [11] explored the electronic word-of-mouth (e-WOM) intentions of Chinese social media influencers regarding artificial intelligence (AI) color cosmetics. The research investigates the factors influencing e-WOM intentions among Chinese social media influencers for AI-based color cosmetics. Findings reveal that product performance and aesthetics significantly impact e-WOM intentions, with influencers primarily valuing aesthetics. The study underscores the growing importance of AI in cosmetics and the need for brands to focus on product aesthetics in influencer marketing. The research does not delve deeply into the ethical and social implications of AI-based cosmetics, and it may benefit from a more extensive exploration of the influencers' motivations and relationships with brands.

Philp, Jacobson, and Pancer (2022) [12] delve into the use of computer vision techniques to predict social media engagement, focusing on food marketing content on Instagram. The research employs computer vision to analyze Instagram posts related to food marketing and predicts engagement metrics. Their findings reveal that visual features significantly influence engagement, with image aesthetics and food presentation playing crucial roles. This study underscores the potential of computer vision in enhancing social media marketing strategies. One potential limitation is the study's exclusive focus on food-related content, which limits its generalizability to other industries. Additionally, the research may not fully account for all factors influencing engagement, such as the impact of content captions and user interactions.

Albahli and colleagues (2022) [13] present AEI-DNET, a novel deep learning model designed for stock market predictions using technical indicators. The research introduces a hybrid model combining

DenseNet architecture with an autoencoder to enhance stock market forecasting. The model demonstrates superior predictive accuracy compared to traditional methods, highlighting the potential of deep learning in financial forecasting. However, the paper lacks a comprehensive discussion of model interpretability, which is critical for financial decision-making. Additionally, it might benefit from addressing potential limitations related to data availability and market volatility.

3. PROPOSED SYSTEM

Predicting alpha signals using microblogging data involves several steps, including exploratory data analysis (EDA), dataset preprocessing, feature extraction using TF-IDF (Term Frequency-Inverse Document Frequency), and building a predictive model using XGBoost. Figure 1 shows block diagram of proposed system. Here's a step-by-step guide on how to approach this task:

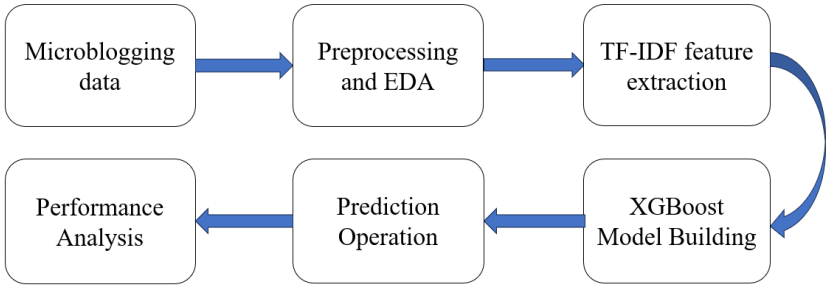

Fig. 1: Block diagram of proposed system.

Step 3: TF-IDF (Term Frequency-Inverse Document Frequency): TF-IDF is a technique to convert text data into numerical features that can be used in machine learning models. We calculate TF-IDF scores for each term (word) in the microblogging dataset. TF-IDF gives higher weights to terms that are frequent within a document but rare across the entire dataset. We can use libraries like scikit-learn to perform TF-IDF vectorization.

Step 4: Splitting the Dataset: We split our preprocessed dataset into training and testing sets. A common split ratio is 80-20, depending on the dataset size.

Step 5: Building an XGBoost Model: We define our XGBoost model by specifying hyperparameters. Some important hyperparameters to consider include the learning rate, tree depth, and the number of trees (boosting rounds). We train the XGBoost model using the training dataset.

Step 6: Model Evaluation: We evaluate the performance of our XGBoost model on the testing dataset using appropriate evaluation metrics. For classification tasks, we can use metrics like accuracy, precision, recall, and F1-score. We analyze the model's performance and check if it meets our prediction accuracy goals.

3.1 TF-IDF Feature Extraction

TF-IDF which stands for Term Frequency – Inverse Document Frequency. It is one of the most important techniques used for information retrieval to represent how important a specific word or phrase is to a given document. Let’s take an example, we have a string or Bag of Words (BOW) and we have to extract information from it, then we can use this approach.

Figure 2 shows the TF-IDF feature extraction block diagram. The tf-idf value increases in proportion to the number of times a word appears in the document but is often offset by the frequency of the word in the corpus, which helps to adjust with respect to the fact that some words appear more frequently in general. TF-IDF use two statistical methods, first is Term Frequency and the other is Inverse Document Frequency. Term frequency refers to the total number of times a given term t appears in the document

doc against (per) the total number of all words in the document and the inverse document frequency measure of how much information the word provides. It measures the weight of a given word in the entire document. IDF show how common or rare a given word is across all documents. TF-IDF can be computed as $tf * idf$.

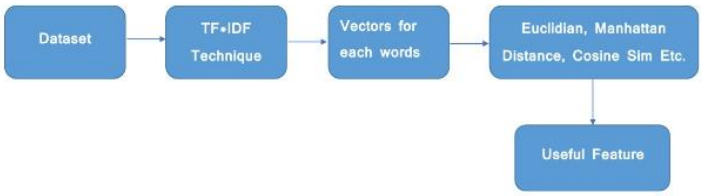

Fig..2: TF-IDF block diagram.

TF-IDF do not convert directly raw data into useful features. Firstly, it converts raw strings or dataset into vectors and each word has its own vector. Then we'll use a particular technique for retrieving the feature like Cosine Similarity which works on vectors, etc.

Terminology

t — term (word)

d — document (set of words)

N — count of corpus

corpus — the total document set

Step 1: Term Frequency (TF): Suppose we have a set of English text documents and wish to rank which document is most relevant to the query, "Data Science is awesome!" A simple way to start out is by eliminating documents that do not contain all three words "Data" is", "Science", and "awesome", but this still leaves many documents. To further distinguish them, we might count the number of times each term occurs in each document; the number of times a term occurs in a document is called its term frequency. The weight of a term that occurs in a document is simply proportional to the term frequency.

$$tf(t,d) = count\ of\ t\ in\ d\ /\ number\ of\ words\ in\ d$$

Step 2: Document Frequency: This measures the importance of document in whole set of corpora, this is very similar to TF. The only difference is that TF is frequency counter for a term t in document d, whereas DF is the count of occurrences of term t in the document set N. In other words, DF is the number of documents in which the word is present. We consider one occurrence if the term consists in the document at least once, we do not need to know the number of times the term is present.

$$df(t) = occurrence\ of\ t\ in\ documents$$

Step 3: Inverse Document Frequency (IDF): While computing TF, all terms are considered equally important. However, it is known that certain terms, such as "is", "of", and "that", may appear a lot of times but have little importance. Thus, we need to weigh down the frequent terms while scale up the rare ones, by computing IDF, an inverse document frequency factor is incorporated which diminishes the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely. The IDF is the inverse of the document frequency which measures the informativeness of term t. When we calculate IDF, it will be very low for the most occurring words such as stop words (because stop words such as "is" is present in almost all of the documents, and N/df will give a very low value to that word). This finally gives what we want, a relative weightage.

$$idf(t) = N/df$$

Now there are few other problems with the IDF, in case of a large corpus, say 100,000,000, the IDF value explodes, to avoid the effect we take the log of idf . During the query time, when a word which is not in vocab occurs, the df will be 0. As we cannot divide by 0, we smoothen the value by adding 1 to the denominator.

$$idf(t) = log(N/(df + 1))$$

The TF-IDF now is at the right measure to evaluate how important a word is to a document in a collection or corpus. Here are many different variations of TF-IDF but for now let us concentrate on this basic version.

$$tf - idf(t,d) = tf(t,d) * log(N/(df + 1))$$

3.2 XGBoost Model

XGBoost is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "XGBoost is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the XGBoost takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

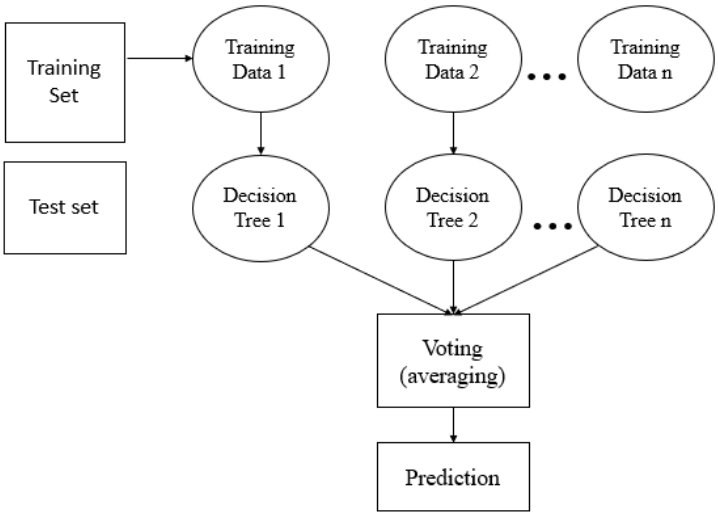

Fig. 3: XGBoost algorithm.

XGBoost, which stands for "Extreme Gradient Boosting," is a popular and powerful machine learning algorithm used for both classification and regression tasks. It is known for its high predictive accuracy and efficiency, and it has won numerous data science competitions and is widely used in industry and academia. Here are some key characteristics and concepts related to the XGBoost algorithm:

- **Gradient Boosting:** XGBoost is an ensemble learning method based on the gradient boosting framework. It builds a predictive model by combining the predictions of multiple weak learners (typically decision trees) into a single, stronger model.
- **Tree-based Models:** Decision trees are the weak learners used in XGBoost. These are shallow trees, often referred to as "stumps" or "shallow trees," which helps prevent overfitting.

- **Objective Function:** XGBoost uses a specific objective function that needs to be optimized during training. The objective function consists of two parts: a loss function that quantifies the error between predicted and actual values and a regularization term to control model complexity and prevent overfitting. The most common loss functions are for regression (e.g., Mean Squared Error) and classification (e.g., Log Loss).
- **Gradient Descent Optimization:** XGBoost optimizes the objective function using gradient descent. It calculates the gradients of the objective function with respect to the model's predictions and updates the model iteratively to minimize the loss.
- **Regularization:** XGBoost provides several regularization techniques, such as L1 (Lasso) and L2 (Ridge) regularization, to control overfitting. These regularization terms are added to the objective function.
- **Parallel and Distributed Computing:** XGBoost is designed to be highly efficient. It can take advantage of parallel processing and distributed computing to train models quickly, making it suitable for large datasets.

3.3 Advantages of proposed system

The advantages of a proposed system can vary depending on the specific context and goals of the system. However, here are some common advantages that a well-designed and implemented system can offer:

- Improved Efficiency: A well-designed system can automate tasks and processes, reducing manual effort and improving efficiency. This can lead to cost savings and increased productivity.
- Enhanced Accuracy: Automation reduces the likelihood of human errors, leading to more accurate results and data.
- Scalability: A good system can grow and adapt to changing needs and increased workload. It can handle larger datasets or user loads without a significant drop in performance.
- Consistency: Automated systems can consistently apply rules and processes, ensuring uniformity in operations and reducing the risk of inconsistency or bias.
- Reduced Costs: Automation can lead to cost savings by reducing labor costs and minimizing errors that can be expensive to correct.

4.RESULT

Figure 4 shows a sample dataset used for classifying alpha signals. It likely displays a portion of the dataset, showcasing rows and their corresponding attributes (columns).It provides a visual representation of the data, offering a glimpse of the kind of information the model is trained on.

Figure 5 presents a summary of the dataset used for classifying alpha signals. It may include statistics like mean, standard deviation, quartiles, and count for each attribute (column) in the dataset. It gives an overview of the central tendencies and spread of the data, aiding in understanding the dataset's characteristics.

Figure 6 is a count plot of the 'alpha' column of the dataset. It displays the frequency of each unique value in the 'alpha' column. It helps in visualizing the distribution of classes in the target variable, which is crucial for understanding class imbalances.

	Id	date	ticker	SF1	SF2	SF3	SF4	SF5	SF6	SF7	alpha
0	1	21/08/18	$NTAP	-0.628652	0.988891	-0.055714	0.774379	0.551089	-1.329229	-0.995539	2
1	2	11/10/18	$WYNN	1.315786	1.438754	0.187327	0.608933	-1.153030	1.859441	0.730995	3
2	3	21/08/18	$DRI	-1.141388	-1.455016	0.332755	0.674502	0.111326	-0.478597	-1.488157	1
3	4	10/07/18	$ge	-0.054839	-1.454149	-0.162267	-0.681870	0.307869	-0.529987	0.404172	2
4	5	12/09/18	$FE	-0.686366	0.838865	0.073830	0.679024	0.329463	1.262782	-1.024042	2
...	...	...	...	...	...	...	...	...	...	...	...
27001	27002	05/10/18	$RF	-0.946205	1.871952	0.068230	-0.348439	0.439969	0.297584	-0.634398	3
27002	27003	30/07/18	$PG	-0.962175	0.623644	0.468051	0.245506	-0.290927	-0.658470	-1.112317	3
27003	27004	16/10/18	$JCP	1.382757	-1.382645	-0.008343	-0.276788	-0.869303	-1.563029	1.372273	2
27004	27005	27/07/18	$NVDA	1.088894	-1.123395	0.027197	0.914267	-0.680183	-0.375689	0.394336	3
27005	27006	14/10/18	$WBA	-0.637959	0.621395	-0.636104	-0.810184	1.587782	-0.413540	0.101924	2

27006 rows × 11 columns

Figure 4:sample dataset used for classifying alpha signals

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 27006 entries, 0 to 27005
Data columns (total 11 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Id      27006 non-null  int64
 1   date    27006 non-null  object
 2   ticker  27006 non-null  object
 3   SF1     27006 non-null  float64
 4   SF2     27006 non-null  float64
 5   SF3     27006 non-null  float64
 6   SF4     27006 non-null  float64
 7   SF5     27006 non-null  float64
 8   SF6     27006 non-null  float64
 9   SF7     27006 non-null  float64
 10  alpha   27006 non-null  int64
dtypes: float64(7), int64(2), object(2)
memory usage: 2.3+ MB
```

Figure 5: summary of dataset used for classifying alpha signal

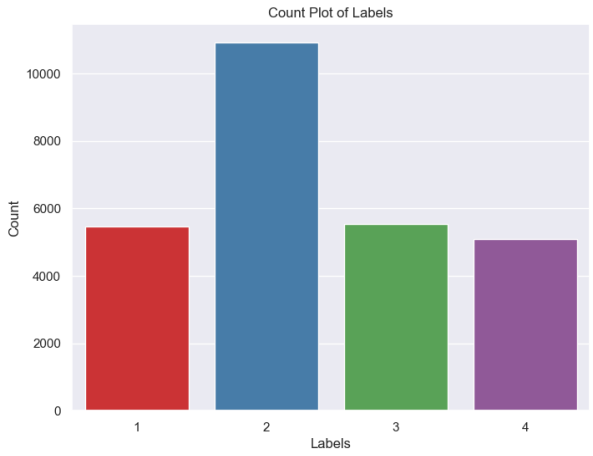

Figure 6: count plot of alpha column of a dataset

Figure 7 shows histograms displaying the distribution of different stock factors (SF1 to SF7). Each histogram represents the frequency of different values for a specific stock factor. It provides insights into the distribution and spread of each stock factor, allowing for a better understanding of their characteristics.

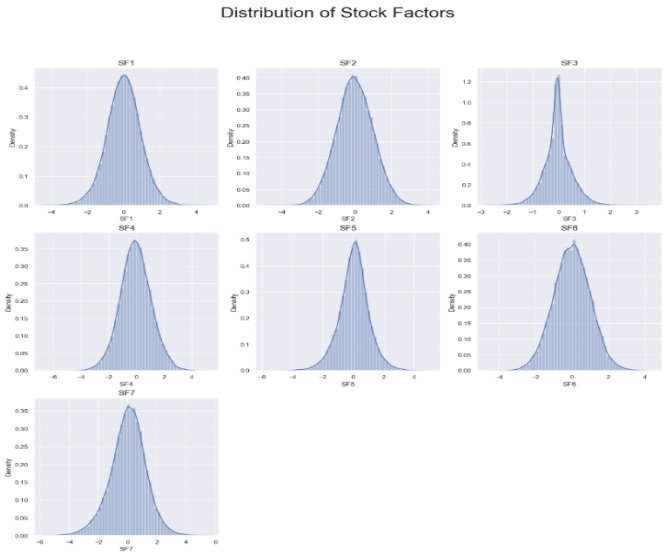

Figure 7: Histogram of distribution of a different stock factor (SF1 to SF7)

Figure 8 contains box plots illustrating the distribution of features in the dataset. Each box plot shows the median, quartiles, and potential outliers for a specific feature.It helps in identifying potential outliers and understanding the spread of the data for each feature.Figu re 9 displays features of the dataset after preprocessing, likely using TF-IDF (Term Frequency-Inverse Document Frequency) transformation. It represents the processed feature set used for model training. It shows the transformed features that the model uses for classification after applying the TF-IDF technique.

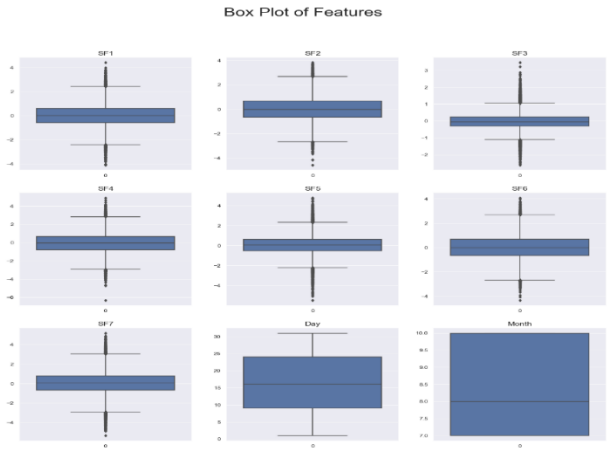

Figure 8: box plots of a features of a dataset

	Id	SF1	SF2	SF3	SF4	SF5	SF6	SF7	Day	Month	...	yelp	yext	yum	zbh	zion	zoes	zto	zts	zumz
0	1	-0.628652	0.988891	-0.055714	0.774379	0.551089	-1.329229	-0.995539	21	8	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	2	1.315786	1.438754	0.187327	0.608933	-1.153030	1.859441	0.730995	11	10	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	3	-1.141388	-1.455016	0.332755	0.674502	0.111326	-0.478597	-1.488157	21	8	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	4	-0.054839	-1.454149	-0.162267	-0.681870	0.307869	-0.529987	0.404172	10	7	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	5	-0.686366	0.838865	0.073830	0.679024	0.329463	1.262782	-1.024042	12	9	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
27001	27002	-0.946205	1.871952	0.068230	-0.348439	0.439969	0.297584	-0.634398	5	10	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27002	27003	-0.962175	0.623644	0.468051	0.245506	-0.290927	-0.658470	-1.112317	30	7	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27003	27004	1.382757	-1.382645	-0.008343	-0.276788	-0.869303	-1.563029	1.372273	16	10	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27004	27005	1.088894	-1.123395	0.027197	0.914267	-0.680183	-0.375689	0.394336	27	7	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27005	27006	-0.637959	0.621395	-0.636104	-0.810184	1.587782	-0.413540	0.101924	14	10	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

27006 rows × 866 columns

Figure 9: Features of A Datset After Preprocessing(Tfidf)

```
0         2
1         3
2         1
3         2
4         2
         ..
27001     3
27002     3
27003     2
27004     3
27005     2
Name: alpha, Length: 27006, dtype: int64
```

Figure 10: Target column of a dataset

```
Classification Report:
              precision    recall  f1-score   support

           1       0.65      0.43      0.52      1097
           2       0.60      0.95      0.73      2153
           3       0.51      0.34      0.41      1114
           4       0.56      0.28      0.37      1038

    accuracy                           0.59      5402
   macro avg       0.58      0.50      0.51      5402
weighted avg       0.58      0.59      0.55      5402
```

Figure 11: Classification report of Decision tree classifier

Figure 10 is a representation of the target column of the dataset. It might show the distribution of different classes in the target variable.It offers insights into the distribution of classes in the target variable, which is essential for understanding class balances or imbalances.

Figure 11 presents the classification report of a Decision Tree classifier. It includes metrics like precision, recall, F1-score, and support for each class. It provides a detailed evaluation of the classification performance of the Decision Tree model.

Figure 12 shows the confusion matrix of the Decision Tree classifier. It visualizes the model's performance by displaying the count of true positive, true negative, false positive, and false negative predictions. It offers a detailed breakdown of the model's predictions, allowing for a comprehensive evaluation of its performance.

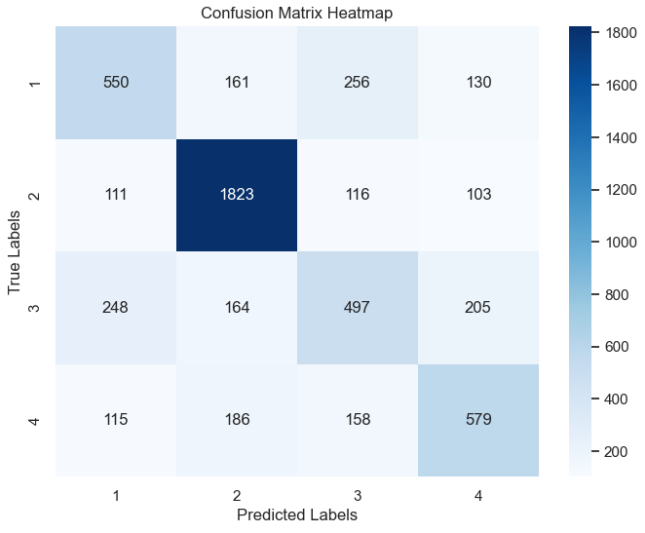

Figure 12: confusion matrix of Decision tree classifier

	precision	recall	f1-score	support
1	0.65	0.43	0.52	1097
2	0.60	0.95	0.73	2153
3	0.51	0.34	0.41	1114
4	0.56	0.28	0.37	1038
accuracy			0.59	5402
macro avg	0.58	0.50	0.51	5402
weighted avg	0.58	0.59	0.55	5402

Figure 13: classification report of XGBoost Classifier

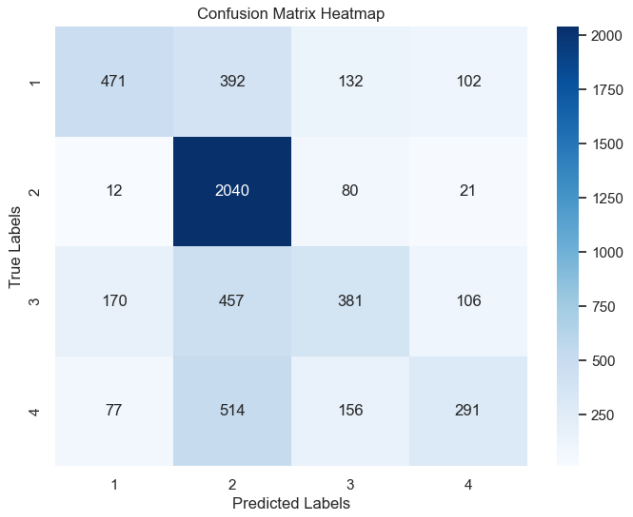

Figure 14: Confusion matrix of XGBoost classifier

Figure 13 contains the classification report of an XGBoost Classifier. Like Figure 8, it provides metrics like precision, recall, F1-score, and support for each class. It offers a detailed evaluation of the classification performance of the XGBoost model.

Figure 14 displays the confusion matrix of the XGBoost classifier. It visualizes the model's performance by showing the count of true positive, true negative, false positive, and false negative predictions for each class. It provides a detailed breakdown of the model's predictions, allowing for a comprehensive evaluation of its performance.

5. CONCLUSION

In conclusion, the process of predicting alpha signals using microblogging data involves a series of well-defined steps, including exploratory data analysis (EDA), dataset preprocessing, feature extraction using TF-IDF, and building a predictive model with XGBoost. EDA is crucial for understanding the characteristics of both the microblogging data and the alpha signals. It helps us gain insights into the dataset's structure and distribution. Dataset preprocessing is essential to ensure data quality. Cleaning and text preprocessing techniques, such as tokenization and TF-IDF, play a significant role in preparing the data for modeling. TF-IDF is a powerful technique for converting text data into numerical features, making it suitable for machine learning models like XGBoost. XGBoost, as an ensemble learning

algorithm, is well-suited for predictive modeling tasks. It can effectively handle both regression and classification tasks and offers various hyperparameters for optimization.

REFERENCES

[1] Ding, Rong, Hang Zhou, and Yifan Li. "Social media, financial reporting opacity, and return comovement: Evidence from Seeking Alpha." Journal of Financial Markets 50 (2020): 100511.

[2] De Choudhury, Munmun, et al. "Predicting depression via social media." Proceedings of the international AAAI conference on web and social media. Vol. 7. No. 1. 2013.

[3] Khaksar Manshad, Mozhdeh, Mohammad Reza Meybodi, and Afshin Salajegheh. "A new irregular cellular learning automata-based evolutionary computation for time series link prediction in social networks." Applied Intelligence 51 (2021): 71-84.

[4] Backstrom, Lars, and Jure Leskovec. "Supervised random walks: predicting and recommending links in social networks." Proceedings of the fourth ACM international conference on Web search and data mining. 2011.

[5] Tur, Benjamin, Jennifer Harstad, and John Antonakis. "Effect of charismatic signaling in social media settings: Evidence from TED and Twitter." The Leadership Quarterly 33.5 (2022): 101476.

[6] Xie, Peng, Hailiang Chen, and Yu Jeffrey Hu. "Signal or noise in social media discussions: the role of network cohesion in predicting the Bitcoin market." Journal of Management Information Systems 37.4 (2020): 933-956.

[7] Kyriazis, Nikolaos, et al. "The differential influence of social media sentiment on cryptocurrency returns and volatility during COVID-19." The Quarterly Review of Economics and Finance 89 (2023): 307-317.

[8] Chandrasekaran, Saravanan, Aditya Kumar Singh Pundir, and T. Bheema Lingaiah. "Deep learning approaches for cyberbullying detection and classification on social media." Computational Intelligence and Neuroscience 2022 (2022).

[9] Pellegrino, Alfonso, Masato Abe, and Randall Shannon. "The dark side of social media: content effects on the relationship between materialism and consumption behaviors." Frontiers in psychology 13 (2022): 870614.

[10] Wang, Tzu-Yin, and Jinah Park. "Destination Information Search in Social Media and Travel Intention of Generation Z University Students." Journal of China Tourism Research (2022): 1-19.

[11] Simay, Attila Endre, et al. "The e-WOM intention of artificial intelligence (AI) color cosmetics among Chinese social media influencers." Asia Pacific Journal of Marketing and Logistics 35.7 (2023): 1569-1598.

[12] Philp, Matthew, Jenna Jacobson, and Ethan Pancer. "Predicting social media engagement with computer vision: An examination of food marketing on Instagram." Journal of Business Research 149 (2022): 736-747.

[13] Albahli, Saleh, et al. "AEI-DNET: a novel densenet model with an autoencoder for the stock market predictions using stock technical indicators." Electronics 11.4 (2022): 611.