Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

Implementation of IndoBERT in Sarcasm Detection using Random Forest Towards Sentiment Analysis

Sabrina Adela Br Sibarani*, Ronsen Purba, Ricky Paian Limbong

Fakultas Informatika, Universitas Mikroskil, Medan, Indonesia
Email: [1,*]sabrina.sibarani@mikroskil.ac.id, [2]ronsen@mikroskil.ac.id, [3]ricky.limbong@mikroskil.ac.id
Correspondence Author Email: sabrina.sibarani@mikroskil.ac.id
Submitted: **21/08/2024**; Accepted: **26/02/2025**; Published**: 01/03/2025**

Abstract−Sarcasm, a subtle form of irony, often introduces a discrepancy between the literal meaning of words and the intended message, making it a significant challenge for sentiment analysis systems. Misinterpreting sarcasm in social media comments can lead to inaccurate sentiment classification, hindering decision-making processes in areas like customer feedback analysis and social opinion mining. This study addresses this issue by evaluating the effectiveness of sarcasm detection in Indonesian text using a Random Forest Classifier (RFC) integrated with IndoBERT. The research employs 10-fold cross-validation to measure performance. Without IndoBERT, the RFC model achieved average accuracy, precision, recall, and F1-score of 78.83%, 78.83%, 79.01%, and 78.83%, respectively. Incorporating IndoBERT significantly improved performance, with all metrics exceeding 84%. Furthermore, 5-fold cross-validation achieved the highest performance, with all metrics reaching 97.24%. This research contributes to developing more robust natural language processing models tailored to Indonesian linguistic contexts, specifically for sarcasm detection.

Keywords: Sarcasm Detection; Random Forest; IndoBERT; Natural Language Processing; 10-Fold Cross Validation

1. INTRODUCTION

Sarcasm involves a divergence between the literal meaning of words and the intended message [1], [2]. As a nuanced form of irony, it is often implicit and commonly employed to convey criticism in a mocking or biting tone [3], [4]. Sarcasm is prevalent in social media comments, where it serves as a means to express emotions and opinions. These comments can provide valuable sentiment data, offering insights into public opinion on various topics such as products, movies, politics, and current events. This information is critical for applications like trend analysis and business strategy development [4] - [8]. However, sarcasm poses a significant challenge for sentiment analysis systems, as its implicit nature can introduce noise, leading to inaccurate results [7]. Numerous researchers have worked to develop models for sarcasm detection to enhance the accuracy of sentiment analysis on social media platforms.

Debby & Auliya (2020) proposed using a Support Vector Machine (SVM) for sentiment analysis and a Random Forest Classifier for sarcasm detection [4]. Their research showed an accuracy of 77.79%, precision of 64.01%, recall of 62.45%, and F1-score of 62.32%. Liu et al. (2024) contributed to the overall predictive performance of the SAHFN-RoBERTa model with an accuracy of 92.73% and an F1-score of 92.63%. However, there are limitations in the dataset that may prevent the model from effectively generalizing its results to different situations [3]. A literature review conducted by Jihad & Ilavarasan (2020) found that sarcasm detection using Bidirectional Encoder Representations from Transformers (BERT) achieved the highest F1 score of 92.4% [1]. MD Saifullah et al. (2023) conducted sarcasm detection using deep learning, and their evaluation showed logistic regression achieved 94% for accuracy, F1-score, and recall, and 95% for precision [9]. Ramisa Anan et al. [9] researched to detect sarcasm in Bengali text by proposing a BERT-based model and applying combined techniques to explain model decisions using Local Interpretable Model-agnostic Explanations (LIME). The resulting model achieved a very high accuracy of 99.60% [10]. Based on studies discussing the use of BERT, the authors see that BERT has the potential to improve the accuracy of sarcasm detection models, as done by Debby & Auliya (2020).

BERT is a natural language processing model that uses a transformer architecture [11], [12]. BERT uses bidirectional representation and considers the context of words in text from both directions. This allows BERT to understand the meaning and context of words better. With this bidirectional representation, BERT can handle various challenges in NLP, such as understanding complex contexts, interpreting ambiguous sentences, and recognizing intricate patterns in text, making it a highly effective model for various natural language processing tasks [13], [14]. Types of BERT include XLM-R, ArBERT, AraBERT, IndoBERT, AlBERT, RoBERTa, and others [15]. This research will use IndoBERT for sarcasm detection. IndoBERT is a natural language model specifically designed for Indonesian [11]. In research conducted by Sani et al. (2022), IndoBERT was used for detecting Indonesian fake news and achieved an accuracy of 94.66% [12].

Based on the above description, this research aims to improve sarcasm detection in Indonesian text by leveraging IndoBERT, a natural language processing model specifically designed for Indonesian, in combination with a Random Forest Classifier (RFC). The study aims to evaluate how the integration of IndoBERT enhances sarcasm detection performance in sentiment analysis tasks, thereby addressing a critical gap in existing NLP tools for the Indonesian linguistic context.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

2. RESEARCH METHODOLOGY

2.1 Research Stages

The stages to be carried out in this research consist of 5 phases, as shown in Figure 1.

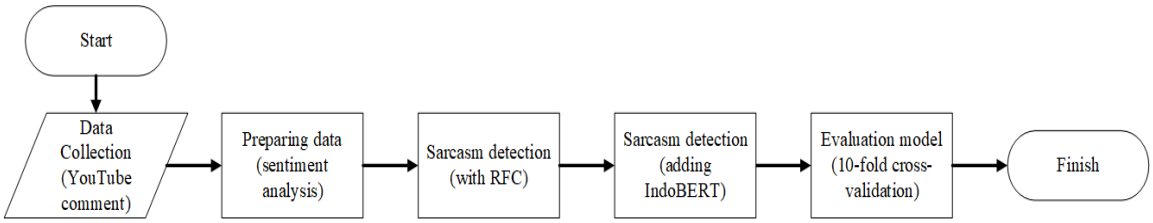

Figure 1. The research stages

2.2 Sentiment Analysis

Sentiment analysis, a subfield of natural language processing (NLP), involves examining textual data to identify the sentiment conveyed by the author. This analysis categorizes text as positive, negative, or neutral based on language patterns and contextual elements. Sentiment analysis has been applied across various domains, including product reviews, movie critiques, political discourse, and discussions of current events [5], [8], [16].
Two methods can be used to train sentiment analysis models [17]:

a. Lexicon - based
   The lexicon-based method utilizes predefined dictionaries or lexicons to identify sentiment or other textual characteristics. Each word in a text is assigned a polarity score based on its inherent meaning or emotional tone. For example, words like "happy" and "love" carry positive polarity scores, while terms like "hate" and "sad" have negative scores. The cumulative polarity score of the text is calculated by summing the individual scores of all words, which is then used to determine the overall sentiment. A positive score reflects a favorable sentiment, a negative score indicates an unfavorable sentiment and a neutral score signifies a neutral sentiment. Lexicon-based approaches are particularly effective for analyzing sentiment in large datasets, such as social media posts or customer feedback.
b. Machine learning
   Machine learning approaches involve the extraction of feature vectors to perform sentiment classification. This process typically encompasses several key phases, including data collection and preprocessing, feature extraction, model training, and result analysis. The machine learning approach requires dividing the dataset into training and testing subsets. The training set enables the model to learn the features of the text, while the testing set is used to assess the model's classification accuracy. This approach is visually represented in Figure 2, which outlines the machine-learning process.

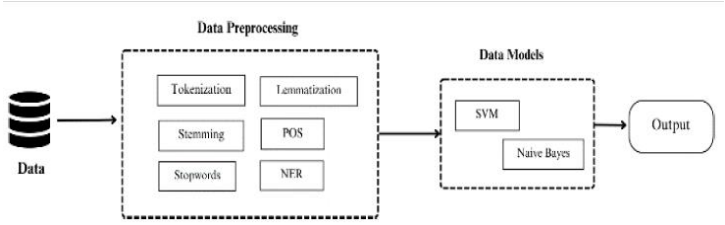

Figure 2. Preprocessing for sentiment analysis using machine learning

2.3 Sarcasm Detection

Sarcasm is a complex form of communication in which speakers often convey a meaning contrary to what is intended, usually to express criticism or mockery in a humorous or sharp manner [18]. Sarcasm involves the use of words that convey the opposite of their literal meaning, often to express a negative attitude toward people or events [19]. Additionally, specific vocal qualities such as nasal tone, breathiness, and pharyngeal tone, as well as acoustic features like jitter and the harmonic-to-noise ratio, can be associated with sarcastic speech. These features are commonly encountered in everyday language and are often challenging to identify, particularly when expressed in written form or across different languages and cultures [20].

In the context of text, sarcasm can be identified through various linguistic features such as n-grams, with a particular emphasis on pattern-based features that are crucial for its detection. Additionally, lexical cues such as interjections and positive affective terms can serve as indicators of sarcasm. Models incorporating these features have demonstrated the ability to assess sarcasm in a manner comparable to human evaluators [18] - [20].

In general, sarcasm detection involves extracting four types of features: sentiment-related, punctuation-related, lexical and syntactic, and pattern-related [21]

a. Sentiment-related

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

Sentiment-related features pertain to the analysis or mining of sentiments within a text or tweet. This includes identifying emotions or feelings expressed in the writing, whether positive, negative, or neutral. In the context of this study, "sentiment-related" also encompasses the use of words with positive or negative emotional content to convey sarcasm.

b. Punctuation-related
Punctuation-related features are used to detect sarcasm by examining the use of punctuation within a text or tweet. Sarcasm often employs specific punctuation marks or the repetition of vowels to convey a meaning contrary to the literal interpretation or to reflect certain expressions, such as a lowered tone or facial expressions. Thus, features related to punctuation are extracted to aid in identifying sarcasm in text.

c. Lexical dan syntactic.
Lexical features refer to the use of words within a text or tweet. In this context, "lexical" can refer to the use of common or uncommon words, as well as expressions generally associated with sarcasm. For instance, complex sentences or unusual words might indicate sarcasm by creating ambiguity for the reader or listener. Meanwhile, syntactic features pertain to the structure or arrangement of words within a sentence or text. In this context, "syntactic" refers to the use of complex sentences or unconventional sentence patterns to obscure the true meaning of a statement. For example, unusual sentence structures or complicated sentences can signal sarcasm, as they may be used to mask the speaker's true feelings or opinions.

d. Pattern-related
Pattern-related features refer to aspects of text analysis related to patterns of specific words or phrases often associated with sarcasm. These patterns are selected from common sarcastic expressions as they frequently appear in everyday conversations, both spoken and written. Although not numerous, these patterns tend to indicate sarcasm. However, since many tweets in the training and testing data do not contain these patterns, further research is conducted to extract additional feature sets. Pattern-related features are inspired by the approach used by Davidov et al., where words are classified into two categories based on their frequency in the dataset: high-frequency words and content words. Patterns are then defined as regular sequences of high-frequency words and slots for content words in a text, which are used to identify sarcasm.

2.4 IndoBERT

BERT (Bidirectional Encoder Representations from Transformers) is a natural language processing model that employs transformer architecture [11], [12]. BERT leverages transformers—an attention mechanism that captures contextual relationships between words (or subwords) in a text. The transformer architecture consists of two components: an encoder, which processes the input text, and a decoder, which generates predictions for a specific task. Since BERT is intended to produce a language model, only the encoder is used. Unlike sequential models that process input text in a fixed order (either left-to-right or right-to-left), the transformer's encoder processes the entire sequence of words simultaneously. This allows the model to grasp the context of a word based on its surrounding words (both left and right) [11], [12] With its bidirectional representation, BERT effectively addresses challenges in NLP, such as interpreting complex contexts, resolving ambiguous sentences, and recognizing intricate patterns in text, making it a robust model for a wide range of natural language processing tasks [13], [14]. BERT operates with two distinct frameworks: pretraining and fine-tuning, as illustrated in Figure 3.

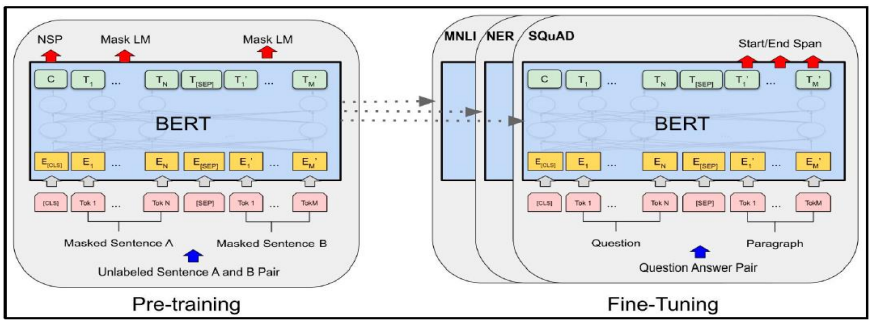

Figure 3. Pre-training and fine-tuning pada BERT [22]

Koto et al and Sani et al [11], [12] characterize IndoBERT as a BERT model tailored for Indonesians, trained through masked language modeling on a dataset composed of Indonesian text. IndoBERT consists of 12 hidden layers. In its word embedding process, IndoBERT starts by adding special tokens, [CLS] and [SEP], at the beginning and end of each sentence to mark these boundaries. The subsequent step is tokenization, which decomposes sentences into individual words or tokens. This tokenization uses the word-piece method, integrating both the model's internal vocabulary and any out-of-vocabulary terms. The model's vector representations are fixed at a dimension of 768. The tokenized words are then mapped to the vocabulary in the corpus, transforming each sentence into a sequence of 12 tokens, with each token represented by 768 distinct dimensions.

Prior to feature extraction, segment differentiation and position embedding are performed to enhance the model's understanding of context. Segment differentiation involves vector representations being applied solely at

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

index 0 and index 1. When the input comprises a single sentence, the segment embedding is set to index zero. Additionally, position embedding utilizes a lookup table of size (n, 768), where \(n \) denotes the sentence length. The first row represents the vector representation for words in the first position, the second row for the second position, and so forth. The integration of these three embedding processes is referred to as input embedding, which enables the pre-trained model to better adapt to various Natural Language Processing (NLP) tasks.

2.5 Random forest classifier

Random Forest is a supervised machine learning algorithm that extends decision tree methods to handle regression and classification problems. It employs ensemble learning, to integrate multiple classifiers to address complex challenges. The Random Forest algorithm consists of numerous decision trees, and the "forest" it generates is trained using a method known as bagging, or bootstrap aggregating. Bagging is an ensemble meta-algorithm that enhances the accuracy of machine-learning models [23].

In Random Forest classification, the ensemble approach is applied to achieve precise results. This involves training multiple decision trees with a training dataset. Each dataset used for training includes various observations and features that are randomly selected during the node-splitting phase. The strength of Random Forest lies in the collective operation of these decision trees, each structured with decision nodes, leaf nodes, and a root node. Predictions from each tree are derived from its leaf nodes. The final prediction of the Random Forest model is determined by aggregating the outputs from all trees through a majority voting system. In essence, the final prediction represents the most frequently occurring outcome across all the decision trees in the forest. The classifier structure of the Random Forest is depicted in a simplified form in Figure 4.

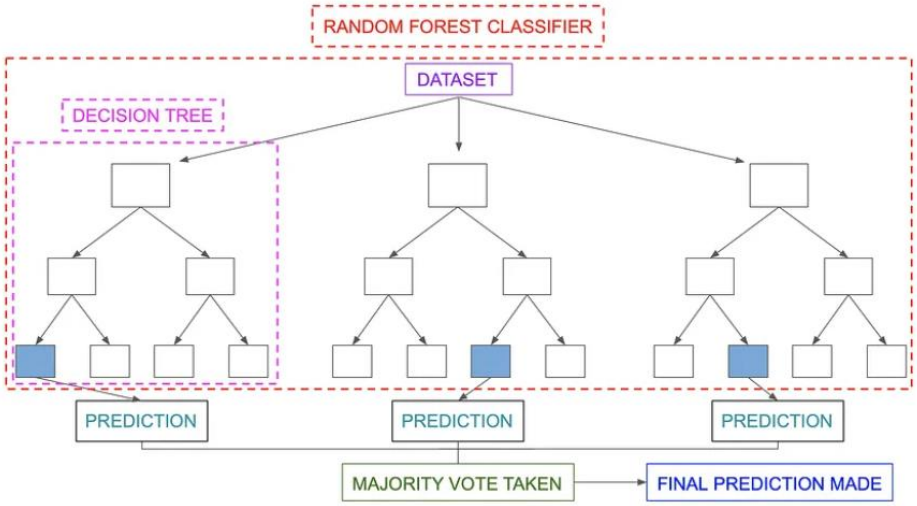

Figure 4. Random forest classifier

The Random Forest Classifier (RFC) stage in this study is primarily used for classification, specifically to predict whether a comment is sarcastic or not based on features extracted from the text. In this context, RFC is a machine learning algorithm used to predict the label (Is_Sarcastic), utilizing a variety of input features derived from the comments in the dataset. In this case, RFC is used for Classification Task, Extracted Features, and Ensemble Learning.

a. Classification task
   1. The primary role of the Random Forest in this model is to classify whether a comment is sarcastic or not based on the feature set provided.
   2. The target variable (y in the code), Is_Sarcastic, represents the binary labels (sarcastic vs. non-sarcastic), which the Random Forest classifier will try to predict.

b. Extracted Feature
   These features are based on the textual content of the comments. The RFC model requires numerical features to perform classification, so these features are extracted and transformed into numerical vectors. The code uses the TF-IDF Vectorizer (which is part of the preprocessing step) to convert the textual data into numerical vectors representing the comments. These vectors capture the relative importance of words in the document and help the model distinguish different patterns in the text.

c. Ensemble Learning
   Random Forest is an ensemble method that constructs multiple decision trees (100 trees in this case) and combines their predictions. Each decision tree is trained on a random subset of the training data and uses different features to make decisions. This helps the model capture complex patterns and reduces overfitting. The final output of the Random Forest classifier is determined by majority voting across all the trees, where the label (sarcastic or not sarcastic) predicted by most trees is selected as the final prediction.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

2.6 K-fold cross-validation

Cross-validation is a statistical method commonly used in machine learning to compare and select models for predictive modeling problems. It is valued for its simplicity, ease of implementation, and broadly lower bias in skill estimates compared to other methods. Common types of cross-validation include K-Fold Cross-Validation, Leave-One-Out Cross-Validation (LOOCV), Stratified K-Fold Cross-Validation, and Leave-P-Out Cross-Validation (LPOCV).

In this study, K-Fold Cross-Validation was employed for model training and evaluation. This procedure involves dividing the dataset into \(k \) groups or folds. A specific value of \(k \) (e.g., \(k = 10 \) for 10-fold Cross-Validation) determines how the data is partitioned. The process uses a limited sample to estimate how the model is expected to perform on unseen data, thereby providing a more general evaluation of the model's performance. K-Fold Cross-Validation is used for model evaluation rather than training. The data distribution for training and testing using 10-fold Cross-Validation is illustrated in Figure 3.

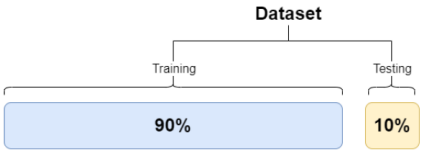

Figure 5. 10-fold Cross Validation [24]

3. RESULT AND DISCUSSION

3.1 Data Collected

Social media is an online platform that enables users to create, share, and exchange information and ideas within virtual communities and networks [25]. These platforms play a crucial role in facilitating instant communication and interaction among individuals, groups, and organizations. Additionally, social media allows users to connect with others, build relationships, and establish professional contacts [26]. As a result, social media plays a significant role in sentiment analysis research, particularly in sarcasm detection [20], [21]. Types of social media platforms include social networking sites like Facebook, Twitter, and LinkedIn, as well as media-sharing networks such as Instagram and YouTube, which focus on sharing photos, videos, and other media [25], [26].

The data used in this research is text data from comments collected from several videos on the social media platform YouTube, with podcast 0063ontent using Google Sheets and web scraping tools. YouTube was chosen as the source of text data for this study because YouTube comments offer a vast amount of data reflecting real-world opinions and reactions on various topics. This data is diverse, covering a wide range of subjects and viewpoints. Data collection was carried out using web scraping techniques with the help of Google Apps Script from a GitHub repository owned by MAN1986 [8]. YouTube video IDs are copied and then placed into the first cell of a Google Sheets document. The code is then pasted into the Google Script, enabling Advanced Google Services – YouTube Data API v3. When the code is run, the scraped results are saved on the second sheet of the open spreadsheet document.

3.2 Preparing Data

The collected data, in the form of a .csv (comma-separated values) document, will proceed with data preparation by conducting sentiment analysis using a support vector machine, as shown in Figure 6.

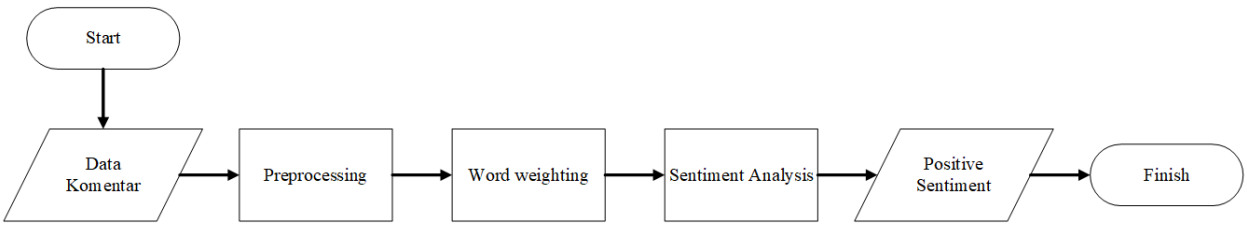

Figure 6. Sentiment analysis process

To perform sentiment analysis, several data preparation stages are necessary for labeling the sentiment of the comments data. These stages include Data Preprocessing, Word Extraction and Weighting, and Sentiment Labeling. Below is a more detailed explanation of each stage:

a. Preprocessing
   In this stage, data preprocessing is carried out to prepare the dataset for further analysis. The preprocessing steps include filling in missing data, removing irrelevant columns, case-folding, removing unnecessary characters and words, tokenization, stemming, and word normalization as shown in Figure 7.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

	Comment	Comment_CaseFolded	comment_tokens	comment_tokens_fdist	comment_tokens_WSW	comment_tokens_stemmed	comment_text
0	Saya lebih setuju. Kawasan tambang di kelola u...	saya lebih setuju kawasan tambang di kelola un...	[saya, lebih, setuju, kawasan, tambang, di, ke...	{'saya': 1, 'lebih': 1, 'setuju': 1, 'kawasan'...	[setuju, kawasan, tambang, kelola, penghijauan...	[tuju, kawasan, tambang, kelola, hijau, eru, r...	tuju kawasan tambang kelola hijau eru rusa cua...
1	Musyaearah bikin poling dong next ketua KPK si...	musyaearah bikin poling dong next ketua kpk si...	[musyaearah, bikin, poling, dong, next, ketua,...	{'musyaearah': 1, 'bikin': 1, 'poling': 1, 'do...	[musyaearah, bikin, poling, next, ketua, kpk, ...	[musyaearah, bikin, poling, next, ketua, kpk, ...	musyaearah bikin poling next ketua kpk ajabrda...
2	The tendency of them to ~~most likely~~...	the tendency of them to delmost likelydel supp...	[the, tendency, of, them, to, delmost, likelyd...	{'the': 1, 'tendency': 1, 'of': 1, 'them': 1, ...	[tendency, them, delmost, likelydel, support, ...	[tendency, them, delmost, likelydel, support, ...	tendency them delmost likelydel support each o...
3	asikk bgt bahasanyyaaa ini<a href="UCkszU2WH9g...	asikk bgt bahasanyyaaa inia hrefuckszuwhgymbdv...	[asikk, bgt, bahasanyyaaa, inia, hrefuckszuwhg...	{'asikk': 1, 'bgt': 1, 'bahasanyyaaa': 1, 'ini...	[asikk, bahasanyyaaa, inia, hrefuckszuwhgymbdv...	[asikk, bahasanyyaaa, inia, hrefuckszuwhgymbdv...	asikk bahasanyyaaa inia hrefuckszuwhgymbdvujgx...
4	Mbak Nana, pliss bahas ttg sistem pendidikan k...	mbak nana pliss bahas ttg sistem pendidikan ki...	[mbak, nana, pliss, bahas, ttg, sistem, pendid...	{'mbak': 2, 'nana': 1, 'pliss': 1, 'bahas': 1,...	[mbak, nana, pliss, bahas, sistem, pendidikan,...	[mbak, nana, pliss, bahas, sistem, didik, kita...	mbak nana pliss bahas sistem didik kitaa mbak

Figure 7. Preprocessing result

b. Word Extraction and Weighting
After data cleaning, the next step is word extraction and weighting using TfidfVectorizer. At this stage, TfidfVectorizer calculates the TF-IDF weight for each word in each document to indicate the importance of a word in that document and the overall corpus. Words that appear more frequently in a specific document but rarely in others will have a high TF-IDF weight. Figure 4 shows the sparse matrix representation of the TF-IDF generated by TfidfVectorizer.

```
  (0, 2768)     0.31177554513867645
  (0, 1993)     0.23541390954030253
  (0, 1712)     0.5952560173467816
  (0, 1417)     0.5650729384683119
  (0, 822)      0.3763009975937094
  (0, 328)      0.1792671411097024
```

Figure 8. TFidfVectorizer result

c. Sentiment labeling
After word weighting, sentiment labeling is applied to the comments data using a pre-trained sentiment classification model. The result of this sentiment labeling, particularly positive sentiment, will be used for sarcasm detection. The sentiment labeling process identified 3,618 positive sentiment data points, as shown in Figure 9.

	Comment	Sentiment
0	Saya lebih setuju. Kawasan tambang di kelola u...	negative
1	Musyaearah bikin poling dong next ketua KPK si...	neutral
2	The tendency of them to ~~most likely~~...	negative
3	asikk bgt bahasanyyaaa ini<a href="UCkszU2WH9g...	neutral
4	Mbak Nana, pliss bahas ttg sistem pendidikan k...	neutral

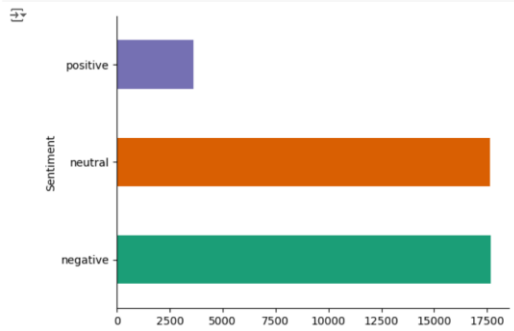

Figure 9. Sentiment analysis result

3.3 Sarcasm Detection

This study aims to develop a highly accurate model capable of distinguishing between sarcastic and non-sarcastic sentences within a comment dataset. The model was trained on a dataset of 3,618 positively labeled comments obtained through a prior sentiment analysis process. To achieve this, we employed a random forest classifier, extracting sentiment-related, punctuation-related, lexical, and syntactic features. The process of sarcasm detection using RFC is illustrated in Figure 10.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

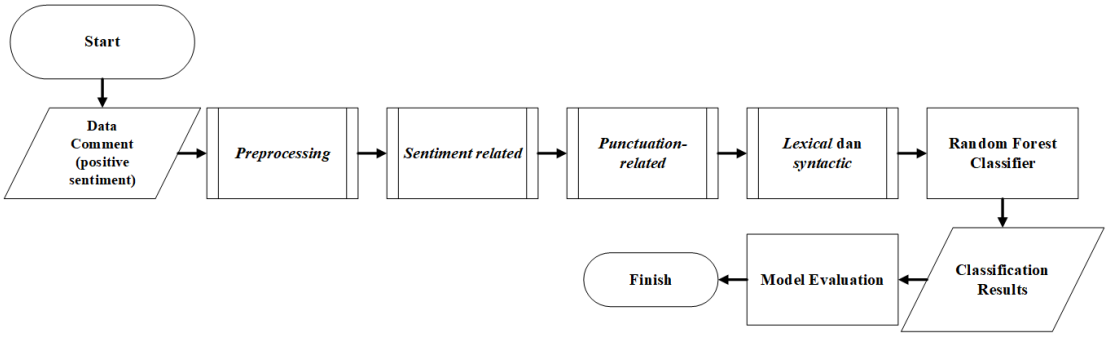

Figure 10. Sarcasm detection with RFC process

a. Sentiment related
In sentiment-related analysis, VADER (Valence Aware Dictionary and Sentiment Reasoner) was employed to extract sentiment-related features. VADER is specifically tailored to process informal language commonly encountered on social media, making it particularly effective for texts that include slang, emoticons, and abbreviations. It generates four sentiment scores: negative (indicating the degree of negativity), neutral (reflecting the level of neutrality), positive (demonstrating the extent of positivity), and compound (a composite score representing the overall sentiment, with a range from -1 (most negative) to +1 (most positive)). The results of the sentiment-related analysis using VADER are shown in Table 1.

Table 1. Sentiment related result

	Comment	vader_neg	vader_neu	vader_pos	vader_compound
0	Sharp like it bukan mempermuda tp mempermudah	0	0.706	0.294	0.3612
1	Biaya jadi cagub, walikota, dan pemerintahan lainnya itu muahal Makanya pas kepilih, orientasinya cari keuntungan lagi Bukan memajukan daerah Soal kecerdasan, kemajuan, konsep itu urusan belakangan	0	1	0	0
...	...	...	...	...	...
3616	asssswwwww aswww lucu bgt wkwkkw	0	1	0	0
3617	Jangan !!!!! Jangan ragu ragu	0	1	0	0

b. Punctuation related
In the punctuation-related analysis, features associated with punctuation were extracted by counting various punctuation marks in the comments. This process was carried out using the `extract_lexical_syntactic_features` function, and the results are presented in Table 2.

Table 2. Punctuation related result

	Comment	exclamation_ count	question_ count	all_caps _count	quote_ count	repeated_vowels _count
0	Sharp like it bukan mempermuda tp mempermudah	0	0	0	0	0
1	Biaya jadi cagub, walikota, dan pemerintahan lainnya itu muahal Makanya pas kepilih, orientasinya cari keuntungan lagi Bukan memajukan daerah Soal kecerdasan, kemajuan, konsep itu urusan belakangan	0	0	0	0	0
...	...	...	...	...	...	...
3616	asssswwwww aswww lucu bgt wkwkkw	0	0	0	0	0
3617	Jangan !!!!! Jangan ragu ragu	5	0	0	0	0

c. Lexical and syntactic
Lexical and syntactic feature extraction was performed by counting specific patterns in the text. The steps included counting the number of emojis, the number of repeated vowels, the number of laughing words, and the number of rare words (more than 7 characters). The results of the lexical and syntactic feature extraction are shown in Table 3.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

Table 3. Lexical and syntactic result

	Comment	laughter_ count	exclamation_ countv2	word_ count	rare_word _count
0	Sharp like it bukan mempermuda tp mempermudah	0	0	7	0
1	Biaya jadi cagub, walikota, dan pemerintahan lainnya itu muahal Makanya pas kepilih, orientasinya cari keuntungan lagi Bukan memajukan daerah Soal kecerdasan, kemajuan, konsep itu urusan belakangan	0	0	31	0
...	...	...	...	...	...
3616	asssswwwww aswww lucu bgt wkwkkw	0	0	5	0
3617	Jangan !!!!! Jangan ragu ragu	0	0	9	0

d. Random forest classifier model

1. Training the model
The model is trained using the TF-IDF features (which are based on text patterns like frequency of words, punctuation, sentiment, etc.) and the corresponding sarcastic labels. This process can be seen in Figure 11.

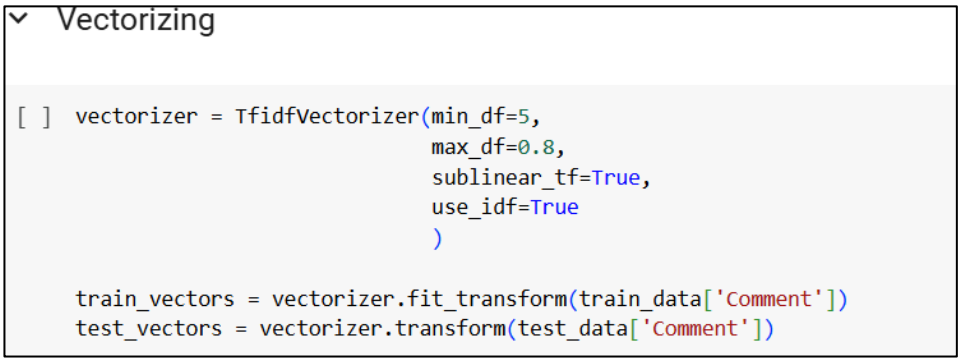

Figure 11. Model training process

2. Making predictions
After training, the model is used to predict sarcasm in new comments by evaluating the same TF-IDF features in the test dataset. The decision trees in the Random Forest will use these features to classify each comment as sarcastic or non-sarcastic. The prediction process can be seen in Figure 12.

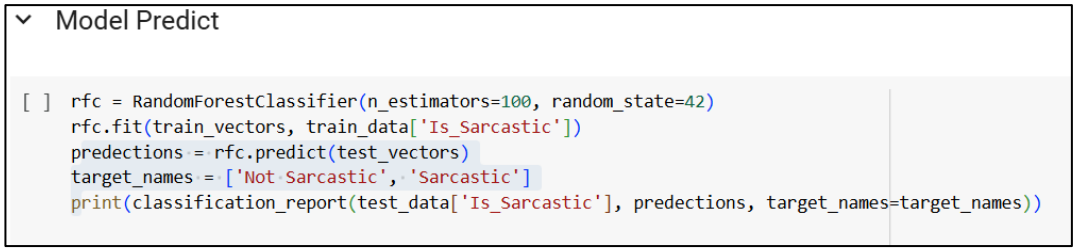

Figure 12. Prediction process

e. Result
After extracting all features, the scores are combined into a Sarcasm_Score, which is then used for sarcasm classification. The resulting dataset contains 1,857 sarcastic labels and 1,761 non-sarcastic labels. The results of the labeling process are presented in Table 4.

Table 4. Sarcasm detection result

	Comment	Sarcasm_score	Is_Sarcastic
0	Sharp like it bukan mempermuda tp mempermudah	0	Not Sarcastic
1	Biaya jadi cagub, walikota, dan pemerintahan lainnya itu muahal Makanya pas kepilih, orientasinya cari keuntungan lagi Bukan memajukan daerah Soal kecerdasan, kemajuan, konsep itu urusan belakangan	0	Not Sarcastic
...	...	...	...
3616	asssswwwww aswww lucu bgt wkwkkw	0	Not Sarcastic

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

3617	Jangan !!!!! Jangan ragu ragu	5	Sarcastic

f. Model performance evaluation
This section will evaluate the performance of the RFC model using 10-fold cross-validation with several evaluation metrics, including accuracy, precision, recall, and F1-score.

3.4 Model Evaluation

This study evaluated the sarcasm detection model using 10-fold cross-validation on accuracy, precision, recall, and F1-score.

a. Random forest classifier (RFC)
Cross-validation provides a more accurate assessment of the model's performance on unseen data during training. The results of the sarcasm detection model evaluation using RFC are presented in Table 5.

Table 5. Evaluation of sarcasm detection model with RFC result

Fold	*accuracy*	*precision*	*recall*	f1
1	0.7818	0.7828	0.7818	0.7821
2	0.7790	0.7791	0.7790	0.7790
3	0.7624	0.7647	0.7624	0.7624
4	0.8177	0.8186	0.8177	0.8175
5	0.7873	0.7873	0.7873	0.7873
6	0.7818	0.7821	0.7818	0.7818
7	0.7983	0.8059	0.7983	0.7994
8	0.8066	0.8069	0.8066	0.8065
9	0.7729	0.7767	0.7729	0.7719
10	0.7950	0.7972	0.7950	0.7952
Rata-rata	0.7883	0.7901	0.7883	0.7883

The model evaluation results presented in Table 5 show that the average values obtained from 1-Fold to 10-Fold cross-validation tests are an accuracy, recall, and F1-score of 78.83%, with a precision of 79.01%. The random forest classifier model achieved its highest accuracy, precision, recall, and F1-score during the 4-Fold test, with an accuracy and recall of 81.77%, precision of 81.86%, and an F1-score of 81.75%.

b. Sarcasm Detection with IndoBERT in RFC
This section will evaluate the IndoBERT model that has been integrated into the RFC model. The evaluation results of the IndoBERT-enhanced RFC model are presented in Table 6.

Table 6. Evaluation sarcasm detection model with IndoBERT in RFC result

Fold	*accuracy*	*precision*	*recall*	f1
1	0.8039	0.8048	0.8039	0.8039
2	0.7514	0.7517	0.7514	0.7514
3	0.8619	0.8668	0.8619	0.8617
4	0.9199	0.9312	0.9199	0.9196
5	0.9724	0.9739	0.9724	0.9724
6	0.8674	0.8958	0.8674	0.8655
7	0.8564	0.8891	0.8564	0.8539
8	0.7956	0.7957	0.7956	0.7956
9	0.7479	0.7512	0.7479	0.7465
10	0.8449	0.8451	0.8449	0.8449
Rata-rata	0.8422	0.8505	0.8422	0.8415

The model evaluation results shown in Table 6 indicate that the average values obtained from 1-Fold to 10-Fold cross-validation tests are an accuracy and recall of 84.22%, precision of 85.05%, and an F1-score of 84.15%. The RFC model with IndoBERT achieved its highest accuracy, precision, recall, and F1-score during the 5-Fold test, with an accuracy, recall, and F1-score of 97.24%, and a precision of 97.39%.

4. CONCLUSION

Based on the results and discussions, the conclusions drawn are that the addition of IndoBERT to the sarcasm detection model with RFC significantly improves sarcasm detection performance. The model evaluation results shown in Table 5 indicate that the average values obtained from 1-fold cross-validation to 10-fold cross-validation are accuracy, recall, and F1-score of 78.83%, and precision of 79.01%. The Random Forest Classifier model evaluation provides the highest accuracy, precision, recall, and F1-score in 4-Fold testing with accuracy and recall at 81.77%, precision at 81.86%, and F1-score at 81.75%. The model evaluation results shown in Table 6 indicate that the average values

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

obtained from 1-fold cross-validation to 10-fold cross-validation are accuracy and recall of 84.22%, precision of 85.05%, and F1-score of 84.15%. The RFC model with IndoBERT provides the highest accuracy, precision, recall, and F1-score in 5-fold testing, with accuracy, recall, and F1-score of 97.24%, and precision of 97.39%. For future research, conducting more intensive hyperparameter searches using grid search or random search on RFC and IndoBERT is recommended to find more optimal configurations. Consider adding other features, such as word embeddings like Word2Vec or GloVe. Expand the dataset with data from various sources and different domains to make the model more generalizable and apply data augmentation techniques to increase training data variability without collecting new data. The IndoBERT-based sarcasm detection model and Random Forest Classifier can be integrated with APIs in social media platforms and customer service, using cloud computing for scalability, and undergoing further testing and validation to ensure optimal accuracy and performance.

REFERENCES

[1] J. Aboobaker and E. Ilavarasan, "A survey on Sarcasm detection approaches," *Indian Journal of Computer Science and Engineering*, vol. 11, no. 6, pp. 751–771, Nov. 2020, doi: 10.21817/indjcse/2020/v11i6/201106048.

[2] P. Verma, N. Shukla, and A. P. Shukla, "Techniques of Sarcasm Detection: A Review," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2021*, Institute of Electrical and Electronics Engineers Inc., Mar. 2021, pp. 968–972. doi: 10.1109/ICACITE51222.2021.9404585.

[3] H. Liu, R. Wei, G. Tu, J. Lin, C. Liu, and D. Jiang, "Sarcasm Driven by Sentiment: A Sentiment-Aware Hierarchical Fusion Network for Multimodal Sarcasm Detection," *Information Fusion*, vol. 108, p. 102353, Aug. 2024, doi: 10.1016/j.inffus.2024.102353.

[4] D. Alita and A. Rahman, "Pendeteksian Sarkasme pada Proses Analisis Sentimen Menggunakan Random Forest Classifier," *Jurnal Komputasi*, vol. 8, no. 2, 2020.

[5] S. K. Alaramma, A. A. Habu, B. I. Ya'u, and A. G. Madaki, "Sentiment analysis of sarcasm detection in social media," *Gadau Journal of Pure and Allied Sciences*, vol. 2, no. 1, pp. 76–82, Jun. 2023, doi: 10.54117/gjpas.v2i1.72.

[6] W. F. Satrya, R. Aprilliyani, and E. H. Yossy, "Sentiment analysis of Indonesian police chief using multi-level ensemble model," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 620–629. doi: 10.1016/j.procs.2022.12.177.

[7] A. C. Băroiu and Ștefan Trăușan-Matu, "Automatic Sarcasm Detection: Systematic Literature Review," Aug. 01, 2022, *MDPI*. doi: 10.3390/info13080399.

[8] R. P. Limbong, R. Purba, and M. F. Pasha, "PEMANFAATAN ANALISIS SENTIMEN DARI ULASAN PRODUK DI YOUTUBE UNTUK PENGEMBANGAN PRODUK BARU," 2024, doi: 10.36418/syntax-literate.v9i7.

[9] M. S. Razali, A. A. Halin, L. Ye, S. Doraisamy, and N. M. Norowi, "Sarcasm Detection Using Deep Learning with Contextual Features," *IEEE Access*, vol. 9, pp. 68609–68618, 2021, doi: 10.1109/ACCESS.2021.3076789.

[10] R. Anan, T. S. Apon, Z. T. Hossain, E. A. Modhu, S. Mondal, and MD. G. R. Alam, "Interpretable Bangla Sarcasm Detection using BERT and Explainable AI," Mar. 2023, [Online]. Available: http://arxiv.org/abs/2303.12772

[11] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," 2020, Online. [Online]. Available: https://huggingface.co/

[12] S. M. Isa, G. Nico, and M. Permana, "IndoBERT for Indonesian Fake News Detection," *ICIC Express Letters*, vol. 16, no. 3, pp. 289–297, Mar. 2022, doi: 10.24507/icicel.16.03.289.

[13] E. Savini and C. Caragea, "Intermediate-Task Transfer Learning with BERT for Sarcasm Detection," *Mathematics*, vol. 10, no. 5, Mar. 2022, doi: 10.3390/math10050844.

[14] G. Z. Nabiilah, S. Y. Prasetyo, Z. N. Izdihar, and A. S. Girsang, "BERT Base Model for Toxic Comment Analysis on Indonesian Social Media," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 714–721. doi: 10.1016/j.procs.2022.12.188.

[15] A. Rahma, S. S. Azab, and A. Mohammed, "A Comprehensive Survey on Arabic Sarcasm Detection: Approaches, Challenges and Future Trends," *IEEE Access*, vol. 11, pp. 18261–18280, 2023, doi: 10.1109/ACCESS.2023.3247427.

[16] N. Majumder, S. Poria, H. Peng, N. Chhaya, E. Cambria, and A. Gelbukh, "Sentiment and Sarcasm Classification with Multitask Learning," *IEEE Intell Syst*, vol. 34, no. 3, pp. 38–43, May 2019, doi: 10.1109/MIS.2019.2904691.

[17] A. Balpande, S. Panditpautra, and R. Nair, "Advancements and Comparative Analysis of Opinion Mining Techniques: A Review of Methods and Algorithms," in *Proceedings of 3rd International Conference on Advanced Computing Technologies and Applications, ICACTA 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICACTA58201.2023.10392801.

[18] R. Filik, A. Ţurcan, C. Ralph-Nearman, and A. Pitiot, "What is the difference between irony and sarcasm? An fMRI study," *Cortex*, vol. 115, pp. 112–122, Jun. 2019, doi: 10.1016/j.cortex.2019.01.025.

[19] C. I. Eke, A. A. Norman, Liyana Shuib, and H. F. Nweke, "Sarcasm identification in textual data: systematic review, research challenges and open directions," *Artif Intell Rev*, vol. 53, no. 6, pp. 4215–4258, Aug. 2020, doi: 10.1007/s10462-019-09791-8.

[20] S. yun Yang, "Listener's ratings and acoustic analyses of voice qualities associated with English and Korean sarcastic utterances," *Speech Commun*, vol. 129, pp. 1–6, May 2021, doi: 10.1016/j.specom.2021.02.002.

[21] Institute of Electrical and Electronics Engineers and PPG Institute of Technology, *Machine Learning based Sarcasm Detection on Twitter Data*. 2020.

[22] M. S. M. Rudwan and J. V. Fonou-Dombeu, "Hybridizing Fuzzy String Matching and Machine Learning for Improved Ontology Alignment," *Future Internet*, vol. 15, no. 7, Jul. 2023, doi: 10.3390/fi15070229.

[23] K. Dhibi *et al.*, "Reduced Kernel Random Forest Technique for Fault Detection and Classification in Grid-Tied PV Systems," *IEEE J Photovolt*, vol. 10, no. 6, pp. 1864–1871, Nov. 2020, doi: 10.1109/JPHOTOV.2020.3011068.

[24] M. I. K. Sinapoy, Y. Sibaroni, and S. S. Prasetyowati, "Comparison of LSTM and IndoBERT Method in Identifying Hoax on Twitter," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 3, pp. 657–662, Jun. 2023, doi: 10.29207/resti.v7i3.4830.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2120−2130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.5801

[25] D. K. Sharma, B. Singh, S. Agarwal, N. Pachauri, A. A. Alhussan, and H. A. Abdallah, "Sarcasm Detection over Social Media Platforms Using Hybrid Ensemble Model with Fuzzy Logic," *Electronics (Switzerland)*, vol. 12, no. 4, Feb. 2023, doi: 10.3390/electronics12040937.

[26] T. A. S. Rohmah and W. Maharani, "Personality Detection on Twitter Social Media Using IndoBERT Method," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 2, pp. 448–453, Sep. 2022, doi: 10.47065/bits.v4i2.1895.

[27] M. Shrivastava and S. Kumar, "A Pragmatic and Intelligent Model for Sarcasm Detection in Social Media Text," *Technol Soc*, vol. 64, Feb. 2021, doi: 10.1016/j.techsoc.2020.101489.

[28] X. Dong, C. Li, and J. D. Choi, "Transformer-based Context-aware Sarcasm Detection in Conversation Threads from Social Media," May 2020, [Online]. Available: http://arxiv.org/abs/2005.11424