Automatic Detection of Framing in Dutch News Articles about COVID-19 Vaccinations with Machine Learning

# Introduction

This report investigates the automatic detection of *framing* in news articles that are related to COVID-19 by means of machine learning. Framing in news media can be defined as “conceptual tools which media and individuals rely on to convey, interpret and evaluate information” (Neuman et al., 1992, p. 60). In other words, the process of framing selects a part of a perceived reality and strengthens its salience to convey a story. One function of frames is to guide the audience in the process of making sense of the flow of information around a topic (Goffman, 1974). Notably, frames have been found to “shape public perceptions of political issues or institutions” (Semetko & Valkenburg, 2000, p. 94). So while providing a path to understanding multiple aspects of a subject or issue, framing can at the same time sculpt the lens through which people perceive it.

During the COVID-19 pandemic, the topic of vaccinations had a polarizing effect on society (Yousuf, 2021). This report aims to gain insight into the role that the media played in the formation of public opinion around vaccinations in the Netherlands. More specifically, we develop a machine learning pipeline that aims to automatically detect framing in vaccination-related newspaper articles. This is done in three steps. Firstly, an annotation scheme is developed for three different types of framing: *Emphasis framing*, *Risk framing*, and *Valence framing*, according to which a dataset of Dutch news articles is annotated. Secondly, a variety of machine learning algorithms and text representation methods are trained on the annotated data. Thirdly, the performance of each experimental setup is evaluated and discussed.

# Background

### Types of framing

[insert description of framing types here]

Not sure if this section is already written somewhere, I’ll leave it blank for now.

### Previous work on automatic frame detection

An example of previous work on the task of automatic frame detection with machine learning is Kroon, van der Meer & Vliegenthart (2022). Similar to the current project, they manually annotated framing to Dutch news articles and compared *dictionary-based analysis* and *unsupervised machine learning* approaches. Dictionary-based analysis uses a predefined lexicon (i.e., dictionary) that decides whether a given text should be categorized under a specific class or frame. Supervised machine learning uses an algorithm to train a classifier, which then predicts the most suitable frame based on patterns it finds in the text. Kroon, van der Meer & Vliegenthart (2022) found that the machine learning approaches consistently outperform the dictionary-based methods on this task.

A machine learning algorithm cannot interpret texts, but rather requires them to be transformed into machine-readable (i.e.) numerical format. There are a variety of methods to numerically represent texts, the quality of which greatly impacts the performance of the classifier (Goodfellow, Bengio & Courville, 2016). Text representation methods vary in the level of semantic information that is encoded. Kroon, van der Meer & Vliegenthart (2022) found that the more semantically complex word embeddings method outperforms simpler methods such as the bag-of-words model. More specifically, word embeddings that are trained on domain-specific data (in this case news articles), rather than off-the-shelve embeddings models, were found to increase the classifier’s performance the most.

### Classification algorithms

**Logistic Regression** is a simple yet effective tool for text classification. It looks at the words and features in your text and figures out how they relate to different categories of media framing. It assigns a probability score to each category, and the category with the highest score is the one it chooses. It's similar to weighing the importance of words in the text as is done with TF\*IDF below. It generally works well for binary classification tasks, and is well-suited for scenarios with limited data, but can have trouble capturing more complex relations between texts and labels.

**Support Vector Machines** (SVM) try to find the best way to separate different categories by drawing a line (or boundary) in the word space. The algorithm finds the place where this line is as far as possible from points from another class, making it very good at forming clear distinctions between categories. It can capture more complex relations than Logistic Regression but is computationally more expensive.

The **Passive Aggressive classifier (PA)** is a quick decision-maker. When presented with a new text sample, it quickly assigns it to an appropriate category. It processes the data sequentially, so it has the ability to adapt and correct any previous classifications. It is often used for tasks that require continuous model updates or scenarios of online learning. The main downside is its difficulty with class imbalances.

### Text representation methods

A **bag-of-words** (BoW) is a simple model that represents a document in terms of the presence of its words in a constructed vocabulary. The vocabulary comprises all words found in a dataset. Each document is assigned a vector with the length of the vocabulary, where a binary indicator marks the presence or absence of the word at that position in the current document. Note that this results in the loss of information on word order. Additionally, words with varying capitalization are treated as distinct, which may lead to potential differences in the representation of otherwise identical words.

As individual texts typically include only a fraction of the vocabulary, a BoW model often results in high-dimensional, sparse vectors. To manage this dimensionality, a common practice is to reduce the vocabulary by filtering out stop words. Stop words are the most frequently occurring words in a language, like "the," "to," and "and," which are present in nearly every text. These words are considered low in information value due to their widespread usage. Another limitation of BoW lies in the incapability to handle either semantic or orthographic similarity.

To mitigate some of these shortcomings, BoW-representations are often enriched with a method called **TF-IDF**, an acronym for Term Frequency-Inverse Document Frequency (Jones, 1972). This method revolves around assigning value to words in a text based on their information content. The underlying idea is that words appearing frequently throughout the entire corpus may not be effective discriminators. By replacing the binary encoders of BoW with the TF-IDF information values, this representation allows for comparisons between texts in terms of similarity. Texts discussing related subjects, or encompassing related frames, often share an overlap in informative words, which is manifested in the similarity of their vectors.

**Embedding representations** are a more sophisticated text representation technique that deals both with high dimensionality and lack of semantic encoding. Based on the assumption that the meaning of words can be derived from their neighboring words (see the distributional hypothesis by Harris (1954), **word embeddings** represent words as vectors that map them to a point in a multidimensional semantic space. The numbers in the vector represent coordinates of where a word belongs in that space. These vector values are commonly learned by neural networks, which use information about how often words and their neighbors appear together. This results in clusters where words in close proximity to each other occur in similar contexts and are thus semantically similar. Reversely, a large distance between two words indicates a high dissimilarity (Mikolov et al., 2013).

Embeddings offer a significant improvement over methods like BoW and TF-IDF because they provide words with meaningful, context-aware representations. To demonstrate this, one common test involves solving analogies using vector arithmetic. Analogies take the form of "a is to b as x is to y." For example, in the analogy "France is to Paris as Italy is to x," we can find x by subtracting the vectors of France and Italy and then adding the vector of Paris. The result will be the vector for Rome. These properties are not explicitly trained into embedding models; they naturally emerge from the way word vectors are designed (Mikolov et al., 2013).

However, there are limitations to the semantic information that embeddings can capture. They may struggle to distinguish between words with opposite meanings, such as "bad" and "good," as these words often appear in similar contexts, resulting in close vector representations. Another drawback is that embeddings cannot effectively represent words that were not part of the training corpus. Moreover, most embedding models are static, meaning they learn global representations of words, where each occurrence of the same word form maps to the same point in the vector space. This makes it impossible for static embeddings to distinguish between the different senses of words with multiple meanings. So-called contextualized embeddings were designed to tackle this problem, where the representation is influenced by the words occurring in its vicinity, thus allowing for different representations of words that can change meaning depending on the context.

This project includes one example of a contextualized embeddings model, namely those provided by the Sentence-BERT model (Reimers and Gurevych, 2019). These **sentence embeddings** have another advantage, namely that they are developed to more accurately represent larger chunks of text. As the name implies, sentence embeddings map entire sentences to a vector space, so that similar sentences have similar vectors, and are thus located closely. Its ability to capture semantic information beyond word-level allows for the encoding of more complexity than word embeddings.

# Methodology

## Data description

The full data set contains 2809 news articles that are related to COVID-19 vaccinations. The articles were published during the corona pandemic in four of the major newspapers in the Netherlands: *de Volkskrant*, *Trouw*, *Algemeen Dagblad*, and *het Parool*, as well as in regional titles. A random selection of 1045 articles are annotated by three researchers, resulting in a training set of 836 and a test set of 209. Table 1 displays the frequency distribution per label in the annotated corpus.

To find out whether there is a significant association between certain frames, a Fisher’s exact test was used. The following frames occur together significantly often: Responsibility and Morality, Responsibility and Opportunity, Responsibility and Human Interest, Human Interest and Opportunity, and Negative and Risk.

|  |  |
| --- | --- |
| **Label** | **Frequency** |
| R-opportunity | 509 |
| E-responsibility | 490 |
| V-positive | 388 |
| E-info & stats | 370 |
| E-human interest | 224 |
| R-balanced | 223 |
| E-conflict | 202 |
| E-economic consequences | 198 |
| V-mod | 153 |
| E-morality | 86 |
| N/A | 79 |
| V-negative | 8 |
| R-risk | 8 |

**Table 1:** Frequency distribution of labels in the annotated corpus.

## Experimental setup

### Classification algorithms and text representation

The experimental setup encompasses the following machine learning algorithms: Logistic Regression, Support Vector Machines (SVM) and the Passive Aggressive classifier (PA). The latter two algorithms were found to perform well on this task by Kroon, van der Meer & Vliegenthart (2022). The text representation methods are bag-of-words (BoW) enriched with TF\*IDF, pre-trained word embeddings, custom word embeddings, and sentence embeddings. The Background section above provides a description of each method. The machine learning pipeline was implemented in Python and can be accessed here [insert github link].

Two types of word embeddings are employed: pre-trained word embeddings from SpaCy, and custom word embeddings that are trained on a large corpus containing texts about COVID-19. The corpus contains news articles, Tweets, transcriptions of parliamentary debates, and subtitles of the Dutch public broadcasting system. Before transformation into embeddings, the texts are converted to lowercase, and all stop words are removed. The vectors for the articles are obtained by averaging over the embeddings of each word in the article.

The sentence embeddings cannot represent texts that are longer than 521 words. Longer texts are represented by chunking the text into smaller pieces and taking the average of the resulting embeddings. This is a common method to deal with the representation of longer texts, although it may lead to loss of more fine-grained information.

|  |  |
| --- | --- |
| **Text representation methods** | **Classification algorithms** |
| Bag-of-words + TF-IDF | Logistic Regression |
| Pre-trained word embeddings | Support Vector Machines (SVM) |
| Custom word embeddings | Passive Aggressive classifier (PA) |
| Sentence embeddings |  |

**Table 2:** Summary of the classification and representation approaches that are included in Experiment 1.

### Experiment 1

The first experiment entails a comparison of the various text representation methods and classification algorithms to establish which combination yields the best results in the task of automatic frame detection. Two types of classifiers are built: binary and multiclass. A binary classifier makes the simple decision of whether a text contains a certain type of framing or not. A multiclass classifier chooses the most likely label from a set of multiple labels. For Valence framing, a multiclass classifier is implemented with four labels: Positive, Neutral, Negative and None. The latter label is applied if no Valence framing is present. Similarly, a multiclass classifier is built for Risk framing with the following labels: Risk, Balanced, Opportunity, and None. A multiclass classifier is suitable for these types because the labels are mutually exclusive: a text that is labeled with positive valence cannot also be labeled with negative valence. In contrast, texts can be described by multiple Emphasis frames, so a multiclass classifier is unsuitable. For each Emphasis frame, a single binary classifier is implemented, resulting in 6 classifiers (one for Responsibility, Conflict, Human Interest, Economic Consequences, Morality, Info & Stats). In total, 8 classifiers are built in this experiment. The performance of the classifiers is evaluated in terms of Precision, Recall and F-score.

### Experiment 2

The process of manually annotating news articles with three types of framing is time-consuming and expensive. Moreover, subtle cues that signal the human annotator that a frame might apply may be lost in the vector representations, especially for longer texts. Experiment 2 explores whether these drawbacks can be overcome by considering alternative structures of the input. Firstly, we investigate the extent to which article titles convey enough information to be classified with the correct frame. Basing the annotations on titles only considerably cuts the time it takes to gather an annotated dataset. Secondly, we examine whether certain labels are more easily detected if the size of the text is decreased. The texts are cut up into chunks of 10 sentences, possibly causing the more subtle cues to surface.

# Results

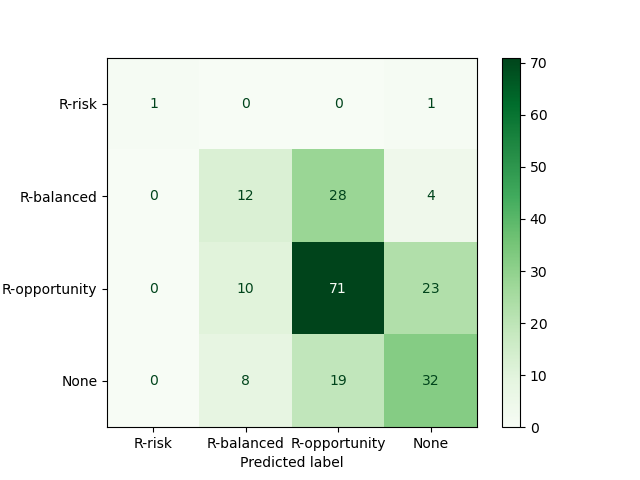
### Experiment 1

This section lists the results of the machine learning experiments. Table 3 displays the evaluation in terms of Precision, Recall and F-score for each label. The system setup with the best scores is included in the table. F-scores higher than 0.60 are considered good enough and are displayed in bold. For these setups and framing labels, it could be expected that the classifier produces reasonable predictions on unseen data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Framing type** | **Setup** | **P** | **R** | **F1** |
| R-risk | SVM/logreg + custom embeddings | **1.00** | **0.50** | **0.67** |
| R-balanced | logreg + custom embeddings | 0.44 | 0.39 | 0.41 |
| R-opportunity | logreg + TF-IDF | **0.56** | **0.88** | **0.69** |
| V-positive | logreg + custom embeddings | 0.64 | 0.53 | 0.58 |
| V-moderate | SVM + custom embeddings | 0.45 | 0.38 | 0.41 |
| V-negative | SVM + sentence embeddings | 0.50 | 0.50 | 0.50 |
| E-responsibility | PA + custom embeddings | **0.61** | **0.87** | **0.72** |
| E-conflict | PA + custom embeddings | **0.63** | **0.57** | **0.60** |
| E-human interest | SVM + word embeddings | **0.75** | **0.77** | **0.70** |
| E-economic | logreg + sentence embeddings | 0.68 | 0.57 | 0.59 |
| E-morality | SVM + custom embeddings | 0.71 | 0.33 | 0.45 |
| E-info & stats | SVM + custom embeddings | **0.80** | **0.65** | **0.72** |

**Table 3**: The best-performing results per label in terms of Precision, Recall and F-score. Only the setup with the best score is displayed.

When interpreting these results, it should be kept in mind that the small sample size can lead to high scores for some of the labels. The Risk label provides a clear example. When inspecting the confusion matrix in Figure 1, it becomes clear that the Risk label occurs twice in the test set, of which one is correctly identified by the SVM classifier. The remaining instance of Risk is categorized as None. Nevertheless, the F-score is high because the classifier achieves a perfect Precision. However, this cannot be regarded as strong or robust evidence that the classifier has learned to identify this label.



**Figure 1**: Confusion matrix for Risk framing predictions by the SVM classifier with custom embeddings. The x-axis represents the predicted labels, and the y-axis the true labels. The numbers correspond to the number of articles in that class.

### Experiment 2

The goal of Experiment 2 is to see if similar or improved scores can be achieved with alternative text inputs. Table 4 displays the results of the classification task with only titles as inputs, both for training and testing. Again, only the best-performing setups are displayed. Only three labels have an F-score higher than 0.60: Opportunity, Responsibility and Info & Stats, which are among the most frequently occurring labels. Table 5 shows the results for paragraphs as input type. This approach has an overlap with the labels that receive an F-score of 0.60 or higher and adds Human Interest to the list.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Framing type** | **Setup** | **P** | **R** | **F1** |
| R-risk | PAC + word embeddings | 0.33 | 0.50 | 0.40 |
| R-balanced | SVM + sentence embeddings | 0.35 | 0.36 | 0.35 |
| R-opportunity | SVM + TFIDF | **0.50** | **0.86** | **0.63** |
| V-positive | PAC + custom embeddings | 0.45 | 0.42 | 0.44 |
| V-moderate | logreg + sentence embeddings | 0.29 | 0.31 | 0.30 |
| V-negative | n/a | 0.00 | 0.00 | 0.00 |
| E-responsibility | PAC + word embeddings | **0.49** | **0.79** | **0.61** |
| E-conflict | SVM + word embeddings | 0.40 | 0.53 | 0.46 |
| E-human interest | PAC + word embeddings | 0.36 | 0.59 | 0.45 |
| E-economic | PAC + sentence embeddings | 0.38 | 0.44 | 0.41 |
| E-morality | PAC + custom embeddings | 0.20 | 0.39 | 0.26 |
| E-info & stats | PAC + word embeddings | **0.54** | **0.77** | **0.64** |

**Table 4:** The best-performing results per label in terms of Precision, Recall and F-score for titles as input type.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Framing type** | **Setup** | **P** | **R** | **F1** |
| R-risk | logreg + sentence embeddings | 1.00 | 0.20 | 0.33 |
| R-balanced | logreg + word embeddings | 0.39 | 0.34 | 0.37 |
| R-opportunity | PA + sentence embeddings | **0.50** | **0.94** | **0.65** |
| V-positive | SVM + word embeddings | 0.53 | 0.66 | 0.59 |
| V-moderate | logreg + sentence embeddings | 0.34 | 0.44 | 0.50 |
| V-negative | SVM + sentence embeddings | 0.17 | 0.35 | 0.39 |
| E-responsibility | SVM + word embeddings | **0.60** | **0.73** | **0.66** |
| E-conflict | PA + custom embeddings | 0.42 | 0.69 | 0.52 |
| E-human interest | SVM + custom embeddings | **0.74** | **0.60** | **0.66** |
| E-economic | logreg + sentence embeddings | 0.42 | 0.44 | 0.43 |
| E-morality | logreg + word embeddings | 0.61 | 0.38 | 0.47 |
| E-info & stats | logreg + custom embeddings | **0.65** | **0.62** | **0.63** |

**Table 5:** The best-performing results per label in terms of Precision, Recall and F-score for paragraphs as input type.

# Discussion

The results of Experiment 1 point out that most Emphasis labels can be decently well predicted by a machine learning classifier. The Responsibility and Info & Stats frames obtain the highest scores, which is in line with expectations because they are the most frequent labels in the dataset. The scores for the Human Interest and Morality frames are sufficient, but the Economic Consequences and Morality frames cannot be accurately detected in this experiment. This is somewhat unsurprising for the Morality frame, as it is relatively rare. However, the Economic Consequences frame was among the easier frames to identify by the human annotators, as a common set of keywords often signals this frame (e.g., mentions of a currency, cafés and restaurants, or entrepreneurs). A simple dictionary approach that includes such keywords can thus be expected to boost the performance for this frame. The same is true for the Morality frame, as it is often evoked by mentions of God, religion or ethics.

Risk and Valence framing prove to be challenging to correctly classify. None of the Valence frames receive a sufficient score, which is unsurprising for Negative and Moderate due to their rarity. Positive, however, is well-represented in the annotations, and could thus be expected to be easily identified. An explanation may be that static word embeddings have difficulty encoding sentiment, as different ends of a polarity spectrum often occur in the same contexts (e.g., “good” and “bad”, “hot” and “cold”). The more advanced contextualized embeddings (in this project sentence embeddings) are better at detecting polarity relations in texts due to their attention to the context. Nonetheless, this is not reflected by the evaluation of the setups, as custom embeddings outperform sentence embeddings on two out of three Valence classes. The use of a vector representation that is enriched with sentiment scores may help the classifier to distinguish between the Valence types.

The only Risk framing class that is easily detected by the classifiers is Opportunity, which is the most frequent label in the dataset. The F-score of Risk is similarly high, but as it is based on only two instances of the class, it should not be regarded as robust evidence. Although Balanced occurs moderately frequently, it is not picked up by the classifier. An explanation might be the subjective nature of the distinction between Balanced and Opportunity. Especially in longer texts, the evidence for Balanced can be scattered and subtle. A brief mention of a risk does not automatically render a text to be balanced if the mention of opportunities is abundant. This distinction has been found to be challenging by the human annotators, which may be reflected by the classifier’s performance on this label.

A common denominator in the best-performing system setups is the presence of custom embeddings. They have been found to yield the best results with relative consistency, which supports the findings by Kroon, van der Meer and Vliegenthart (2022) that training a custom word embeddings model on domain-specific data can significantly boost the classifier’s performance. When another setup outperforms the custom embeddings, it is often a tight race. In terms of algorithms, there is no convincing evidence for a most optimal approach. SVM yields six of the best-performing setups, and Logistic Regression is the best five times. Both are considerably better than the PA classifier, which makes the best predictions for two labels.

Experiment 2 explores whether titles or paragraphs as input instead of full texts can overcome the drawbacks of expensive manual annotations or the loss of subtle information when representing long texts in a machine-readable way respectively. The results provide a clear no: none of the setups outperform the classifiers trained on full texts. For Risk and Valence framing, it does not come as a surprise that the titles do not convey enough information for the classifier, because the cues for these frames often occur later in the texts. However, the human annotators found some of the Emphasis frames to be signaled to some degree by their title, in particular Info & Stats, Responsibility, Human Interest and Conflict. This may not have been true for most texts, since only Info & Stats and Responsibility are detected by the classifiers.

The main limitation of this project lies in the relatively small dataset, which lead to a class imbalance. There is no reliable method to sample the data in such a way that it results in balanced annotations in which the classes occur roughly the same amount. For instance, the Responsibility label is encountered 509 times, providing ample training data for the classifier. Negative valence, however, rarely shows up in the annotations, which renders it unlikely it will be picked up by the classifier. One potential solution to address this challenge is data augmentation, where underrepresented classes can be expanded with artificially generated data. This approach may help bridge the gap between classes, enhancing the classifier's performance.

Another limitation is inherent to the texts, namely that their voluminous nature complicates the detection of frames that are signaled by subtle cues. Splitting the data into smaller segments did not yield improvements, possibly due to the constraint that training could not take place at the paragraph-level without requiring a fresh set of annotations. In Experiment 2, training was conducted on complete texts with their corresponding labels, while predictions were made at the paragraph level, and the evaluation encompassed aggregating the predicted labels across the entire text. An alternative approach could involve annotating the smaller text chunks for training at that level, which might lead to better results. Additionally, a sequence labeling approach, which captures the exact textual evidence evoking a specific frame, could be considered to improve the results.

# Conclusion

In conclusion, these results reveal that machine learning classifiers demonstrated reasonable predictability for most Emphasis labels, particularly Responsibility and Info & Stats, which corresponded with their prevalence in the dataset. However, the challenging Economic Consequences and Morality frames proved more challenging, potentially benefiting from a dictionary approach. Valence and Risk framing exhibited their own complexities, with none of the Valence frames achieving satisfactory scores. This can partially be attributed to the limitations of static word embeddings in representing sentiment, suggesting the potential advantage of sentiment-enriched vector representations. Custom embeddings consistently outperformed other setups, while no single algorithm emerged as the definitive choice.

# References

**Borah, P., Hwang, J., & Hsu, Y. C.** (2021). COVID-19 vaccination attitudes and intention: message framing and the moderating role of perceived vaccine benefits. *Journal of Health communication*, *26*(8), 523-533.

**Goffman, E.** (1974).*Frame analysis: An essay on the organization of experience*. Harvard University Press.

**Goodfellow, I., Bengio, Y., & Courville, A.** (2016). *Deep learning*. MIT press.

**Harris, Z. S.** (1954). Distributional structure. *Word*, *10*(2-3), 146-162.

**Jones, Sparck K.** (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, *28*(1), 11-21.

**Kroon, A. C., van der Meer, T., & Vliegenthart, R.** (2022). B*eyond Counting Words: Assessing Performance of Dictionaries, Supervised Machine Learning, and Embeddings in Topic and Frame Classification*

**Mikolov, T., Chen, K., Corrado, G., & Dean, J.** (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*

**Neuman, W. R., Just, M. R., & Crigler, A. N.** (1992). *Common knowledge: News and the construction of political meaning*. University of Chicago Press.

**Reimers, N., & Gurevych, I.** (2019). Sentence-bert: Sentence embeddings using siamese bert-networks.

**Semetko, H. A., & Valkenburg, P. M.** (2000). *Framing European politics: A content analysis of press and television news.*

**Yousuf, H., Van Der Linden, S., Van Essen, T., Gommers, D., Scherder, E., Narula, J., & Hofstra, L.** (2021). Dutch perspectives toward governmental trust, vaccination, myths, and knowledge about vaccines and COVID-19. *JAMA network open*, *4*(12), e2140529-e2140529.

# Appendix A

[insert full results of all machine learning experiments]