Clark Kent at SemEval-2023 Task 5: SVMs, Transformers, and Pixels for Clickbait Spoiling

Dragos-Stefan Mihalcea
dragos.mihalcea@s.unibuc.ro

Sergiu Nisioi
sergiu.nisioi@unibuc.ro

Human Language Technologies Research Center
Faculty of Mathematics and Computer Science
University of Bucharest

Abstract

In this paper we present an analysis of our approaches for the 2023 SemEval-2023 Clickbait Challenge. We only participated in the sub-task aiming at identifying different clikcbait spoiling types comparing several machine learning and deep learning approaches. Our analysis confirms previous results (Hagen et al., 2022) on this task and show that automatic methods are able to reach approximately 70% accuracy at predicting what type of additional content is needed to mitigate sensationalistic posts on social media. Furthermore, we provide a qualitative analysis of the results, showing that the models may do better in practice than the metric indicates since the evaluation does not depend only on the predictor, but also on the typology we choose to define clickbait spoiling.

1 Introduction

There are many posts in social media that are intended to allure the readers into visiting a specific web page. The phenomenon is commonly known as clickbait and it is usually achieved through sensationalistic formulations that strike curiosity into the readers mind.

Studies show that humans get dopamine rewards during information seeking processes driven by curiosity or pleasure of anticipation (Polman et al., 2022; Kobayashi and Hsu, 2019). Clickbait titles are linguistic exploits that activate these mechanisms and therefore users tend to reduce the perceived information gap (Loewenstein, 1994) by clicking or accessing the advertisements.

While these type of linguistic exploits may be used to steer people into making healthier choices such as avoiding unhealthy foods, making more exercises etc. (Polman et al., 2022), most often, the information overload of clickbait posts from social media poses serios mental health risks (Hwang et al., 2021) and additional risks of propagating misleading information for profit or by the far-right for questionable political purposes (Ecker et al., 2014).

Clickbait spoiling implies rephrasing the original title or the introduction of additional content in the original post with the purpose of limiting the curiosity of the readers. A techno-solutionist pipeline to *spoilling* for social media can be described by the following steps:

1. identify *potential clickbait* posts on social media using text classification models
2. use a bot to access the clickbait websites retrieved at the previous step and extract the corresponding *content* (linked document)
3. identify what *type of information* is required to reduce or limit the degree of curiosity of the original social media post
4. use the *type of information* and the *linked document* identified at the previous steps to generate additional content acting as a spoiler

The first step, for detecting whether a post is clickbait or not, has been greatly explored in recent years including a widely participated Clickbait Challenge in 2017 (Potthast et al.) and several individual studies mentioned here briefly (Vijgen et al., 2014; Blom and Hansen, 2015; Potthast et al., 2016; Indurthi et al., 2020; Mowar et al., 2021).

The task of identifying the different types of spoiler and generating the adequate information for clickbait spoiling (steps three and four) has been tackled more recently with the works of Hagen et al. (2022); Fröbe et al. (2023b) and Johnson et al. (2022), and the 2023 SevEval Task 5 on *Clickbait Spoiling* (Fröbe et al., 2023a).

Our participation at this task is only focused on step number three - classifying different types of clickbait spoilers. In the following sections we will cover a comparison of several linear and deep-learning classifiers together with a quantitative and qualitative analysis of the data.

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1204–1212
July 13-14, 2023 ©2023 Association for Computational Linguistics

Clickbait post	**Spoiler**	**Type**
The deadliest animal in the U.S. may surprise you	Bees, wasps and hornets	phrase
The cheapest place for a last-minute half-term holiday	Cyprus	phrase
Subways are full of bacteria, but here's why you shouldn't freak out	scientists didn't find pathogenic organisms that typically cause sickness	passage
the big failure that hillary kept secret for 30 years	She had failed the D.C. bar exam	passage
Six lessons from the godfather of California cuisine	1) Eat your veggies 2) Enjoy said veggies a few weeks after their season starts. 3) Ingredients dictate everything. 4) Don't serve complex foods to your 4-year-old. 5) You can succeed without a mentor.	multipart
'Scandal' star says she HAS to do this before bed	Bellamy Young meditate	multipart

Table 1: Several examples of clickbait spoiling typology from the data. Type *phrase* are typically short single-word or multi-word expressions, *passage* comprise of longer sentences and *multipart* spoilers require multiple phrases or passages from the original document.

class	#examples	#sentences	avg_sents	std_sents	#tokens	#unq_tokens
phrase	1702	37K	21.5	21.1	706K	360K
passage	1596	45K	27.9	89.1	814K	380K
multi	702	29K	41.9	43.0	529K	227K

Table 2: Dataset statistics computed on the publicly available train/dev splits during the competition. Sentence splitting was done using spacy and only alpha string tokens were taken into account.

2 Background

The SemEval task organizers (Hagen et al., 2022; Fröbe et al., 2023b) have provided a typology of spoilers comprising of three main classes:

1. *phrase* spoilers consisting of a single word or multi-word phrase from the content of the linked document (often consisting of named-entity spoilers)

2. *passage* spoilers consisting of a longer sequence of words or a few sentences from the document

3. *multipart* spoilers consisting of more than one **non-consecutive** phrases or passages from the document

Table 1 contains several examples of clickbait spoiling types together with the accompanying posts. The types *phrase* and *passage* are characterized by their corresponding lengths, where phrase examples are short and passage examples comprising of longer sentences. The type *multipart*, however, is characterized by the lack of cosecutiveness of the phrases or passages comprising the spoiler. A shallow glance over the data reveals that the examples labeled with tag *multipart* share similarities with the other two classes, often times consisting of short phrases or one or two passages. Therefore,

Model	Feature	Dev Acc	Test Acc	%multipart	%passage	%phrase	Acc 5CV
LinearSVC + tf-idf	document	53.4	55.4	26.4	54.9	59.5	46.9
	titles	59.4	57.3	32.2	58.8	62.4	51.1
	post	62.4	61.7	36	62.7	68.1	55.6
SVC +	document	52.7	59.3	24.5	54	60.7	45
n-gram	titles	62	65	20.1	57.5	65.3	51.2
kernel [2 - 8]	post	68.6	65.5	26.8	66.1	70	54.4
SVC +	document	66.7	67.6	0.7	59.9	59.8	39.5
rbf +	titles	62.6	63	25.8	57.1	65.6	49.5
spacy	post	64.7	67.3	29.3	64	66.9	53.4
SVC +	document	60.8	60.4	29.1	59.7	62.2	50.3
rbf +	titles	68.7	67.8	32.5	66.2	67.2	55.3
mpnet-base-v2	post	72.9	72.4	36.6	73.3	68.8	59.6

Table 3: SVM classification results reporting balanced accuracy for different feature types extracted from the linked documents, the document title, and the original post. Columns with % report the average. Acc 5V reports the balanced accuracy across all folds.

one may wonder whether this typology structured in 3 parts is actually consistent given the different criteria that characterize the spoilers.

In Table 2 we present several statistics computed over the train/dev dataset provided during the competition. These are in alignment with the existing data description provided by Hagen et al. (2022) in the paper where the initial version of this dataset was released. Unlike Hagen et al. (2022), we only use the actual content of the linked document to measure the number of sentences, average number of sentences per document and count tokens using spacy[1] (Honnibal et al., 2020). These statistics are computed to identify any class imbalance or structural differences that may exist between the classes in the task.

According to Table 2, the majority of examples require a "phrase" to spoil the clickbait posts. However, the linked documents for these examples are notably shorter and have fewer sentences compared to the "passage" spoilers. This is evident from the larger total number of sentences, higher average number of sentences per document, and greater standard deviation of sentences per document for the "passage" class. It is reasonable to expect this outcome because larger documents inherently need longer passages to reveal the main content of the post.

3 System Overview

Support Vectors

We experiment with several types of support vector machine trained without hyper-parameter tuning. The models use features extracted from the post, title or content of the linked documents:

1. Linear SVC (Vapnik, 1999) with tf-idf features over word unigrams and bigrams as implemented in scikit learn (Pedregosa et al., 2011)

2. SVC with pre-computed n-gram string spectrum kernel (Leslie et al., 2001; Shawe-Taylor et al., 2004), measuring n-grams ranging from 2 to 8; we have implemented the kernel efficiently using scikit-learn tools by employing the hashing trick and cosine similarity over spare Boolean matrices of n-gram occurrences; it is essentially a linear classifier

3. SVC with RBF kernel and spacy (Honnibal et al., 2020) document embeddings from model en_core_web_lg

4. SVC with SentenceTransformers (Reimers and Gurevych, 2019) and sentence similarity model *all-mpnet-base-v2* - based on the pretrained *microsoft/mpnet-base* model fine-tuned using contrastive learning on a 1B sentence pairs dataset[2]

[1] Model name and version: en_core_web_lg==3.5.2

[2] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

Model	Dev Acc	MCC	Test Acc	MCC	%multipart	%passage	%phrase	5-fold CV Acc	MCC
BERT-large-cased	69.3	.51	68.9	.5	52	67.4	70.8	63.4	.47
RoBERTa-large	71.3	.48	70.6	.48	24.2	29.8	87.4	47	.2
DeBERTa	69.7	.53	72	.55	58.1	69.7	73.4	67	.51
PIXEL	63.5	.42	65	.43	47	65.3	68.2	61	.42
SVM (best)	72.9	.44	72.4	.46	36.6	73.3	68.8	59.6	.43

Table 4: Comparison of transformer classification results reporting balanced accuracy for different feature types extracted from the linked documents, the document title, and the original post. Columns with % report the average accuracy per class. Last column reports the average balanced accuracy and Mathews Correlation Coefficient across all folds.

3.1 Text Transformer Models

We test three types of text transformer models (Vaswani et al., 2017) models available from huggingface:

1. BERT large cased (Devlin et al., 2018)
2. RoBERTa large (Liu et al., 2019)
3. DeBERTa large, the first version of the model (He et al., 2021)

We fine-tune these models using the simpletransformers library[3] and identify the best test model based on train-dev evaluation. For cross-validation we train on each split for 10 epochs with learning rate *4e-05*, batch size of 32 and a gradient accumulation of 4 steps.

3.2 Pixel-based Transformer Model

Inspired by recent work of Rust et al. (2023) that addresses the vocabulary bottleneck, we decided to experiment with the Pixel-based Encoder of Language (PIXEL) model for the task of clickbait detection. As the name suggests, the method is based on pixels over characters, literally transforming the texts into RGB image segments of size 16x8464 representing parts of text over a white background using pygame back-end. The pre-trained model, nicknamed PIXEL, is a Vision Transformer Masked Auto Encoder (ViT-MAE (He et al., 2022)) with 112M parameters. The encoder (86M parameters) consists of a 12 layer ViT and the decoder an 8-layer Transformer (26M parameters), the later not being used for downstream tasks.

Training was done for a maximum of 25k steps, with 4 steps for gradient accumulation and an early stopping patience of 5 calls to check when the evaluation worsens. The model stopped after 4500 steps and the total training time took 7 hours, with batch size of 8 per GPU distributed over four Nvidia Tesla V100-SXM2.

While this method is far from optimal with respect to solving the vocabulary bottleneck problem and moves away from the linguistic properties of words, we believed it represents a direction worth testing for a task such as spoiler classification. Its main advantage is the power to generalize across words or languages unseen during training.

4 Experimental Setup

To be consistent with the shared task organizers and to be able to compare against previous work Hagen et al. (2022), we report balanced accuracy for each trained model. Additionally we also measured Mathews Correlation Coefficient (MCC) which appeared to be in-line with the balanced accuracy measures, showing objectively considerably low values between 0.2 and 0.51.

In our experiments, we noticed that the absolute difference between the average Matthews correlation coefficient (MCC) on 5-fold cross-validation and the MCC on the test and development sets was smaller compared to the differences between balanced accuracy scores. The table provided as an example (Table 3) demonstrates that 5-fold cross-validation could be as much as 20 points lower than the accuracy scores on the dev or test sets (last row).

This phenomenon somewhat confirms previous investigations (Chicco and Jurman, 2020) that advocate for MCC being a superior metric over accuracy of F1 based scores. Due to space constraints, we do not report the MCC scores in all our tables, but we indicate the reader to see our reproducible notebooks in the public repository[4].

[3] https://simpletransformers.ai/

[4] https://github.com/mdragos1/

We compared each model against a stratified 5-fold cross-validation approach, where each split included a proportional number of examples from every class. We opted for 5-fold cross-validation to ensure that each split of the combined train (3200) and dev (800) sets from the shared task had similar sizes. We did not use the test set provided by the organizers for cross-validation purposes.

We consider this method to be more appropriate for comparing the models, as it demonstrates their ability to generalize on new data. The SVM model experiments do not benefit from hyperparameter tuning, as we maintained the default parameters specified in scikit-learn, with a regularization constant of $C = 1$.

We conducted experiments using various feature sources for both SVM and deep learning models. Based on our results with SVM classifiers, we found that the original post texts are the most effective source for identifying the type of clickbait spoiling. This finding aligns with similar results reported in previous experiments on the task (Hagen et al., 2022). We can conclude therefore, that the type of spoiler is more of a *characteristic of the post* than a pattern of the linked document or its title.

Additionally, we observed that combining the post text with title or document information only degrades the overall results.

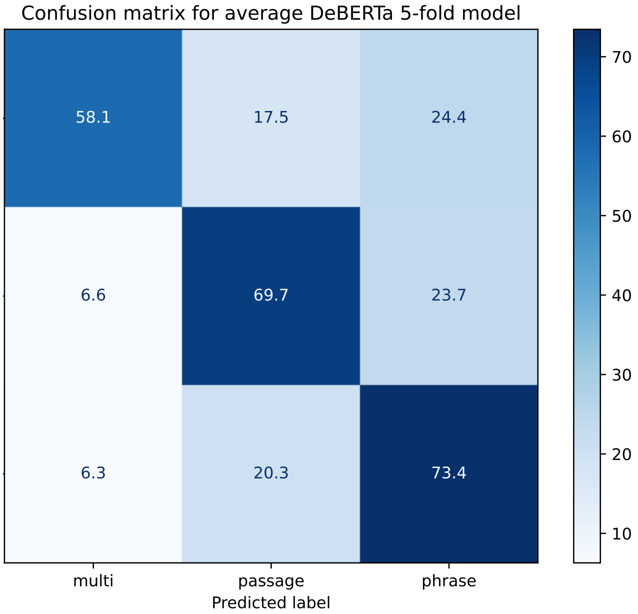

Figure 1: Average confusion matrix of DeBERTa model across all 5 stratified folds.

Overall, for deep learning methods, due to limited access to compute resources, we only employ the post text for classification and not the document or the title.

5 Results

5.1 SVM Classifiers

In this section we summarize our main results using SVM classifiers as resulted from Table 3:

- SVM with string kernel is the most powerful classifier *that uses no other information* than the tokens found in provided the dataset, achieving 65.6 accuracy on the test set without any hyperparameter-tuning; this exceeds the results (59.62) reported by Hagen et al. (2022) where idf scores are extracted from external Open Web Text Corpus (Gao et al., 2020)

- SVM + tf-idf features over word unigrams and bigrams obtains smaller test and dev accuracy and slightly larger than string kernel SVM (but not significantly) 5-fold CV balanced accuracy; neither of these models utilizes any information beyond the tokens present in the dataset; despite the fact that they seem to yield dissimilar results on the test set, we can assert that they exhibit similar performance based on the results obtained from cross-validation

- the results obtained using linear SVC classifiers provide evidence that posts have a shallow grammatical or structural similarity with each other within each class (phrase, passage or multi)

- although SVM with RBF kernel and spacy embeddings yielded high accuracy scores on the dev and test sets, this result appears to be incidental; upon closer examination of the cross-validation results, it is evident that the models do not perform well, and the differences between the CV and test sets can be as much as 20 points lower (39.5 balanced accuracy CV compared to 67.6 on the test set; 0.13 average MCC CV compared to 0.18 on the test set).

- SVM with SentenceTransformers (Reimers and Gurevych, 2019) are the most competitive models, achieving the highest cross-validation scores with the advantage of being the faster

SemEval-Task5-2023

to train and more lightweight than fine-tuning full transformer models

- 5-fold cross-validation is a more reliable way of comparing models, given the small size of the dataset

5.2 Transformer Models

According to Table 4, transformer-based models like DeBERTa and BERT large achieve the most accurate classification results. While the dev/test results appear similar to those obtained by SVMs, closer examination of the cross-validation scores reveals a significant difference between these models. It is not unexpected to observe models that may have been exposed to sections of the data during their pretraining outperforming SVM classifiers that rely on string similarity.

- Table 4 indicates that transformer models perform better than SVM models, particularly when examining the 5-fold cross-validation average; the superiority of transformer models on this task aligns with the findings of Hagen et al. (2022), however, when looking solely at the test set results, SVM models appear to have stronger balanced accuracy and lower MCC

- Table 4 further confirms previous work (Chicco and Jurman, 2020) on the advantages of evaluating using Matthews Correlation Coefficient - being better correlated with the average cross-validation scores

- the best model, according to CV and MCC scores is DeBERTa followed by BERT-large cased with slightly lower values than the ones reported by Hagen et al. (2022), possibly due to different hyper-parameter settings

- PIXEL-based language models (Rust et al., 2023) are comparable with SVM with sentence transformers; these types of models have the power to generalize across new languages, alphabets or even modalities

- according to Figure 1 the majority of confusions stem from the multipart class and rightly so - the class is defined by the existence of discontinuous passages and phrases within the document that can spoil the clickbait post; it is a demanding task for any classifier to identify such traits by looking only at the post text

- we could not run a proper cross-validation experiment using RoBERTa large because it over-fits on certain stratified splits and ends up predicting only the *phrase* label

- the types of clickbait spoilers labeled as phrase and passage leave linguistic traits in the post text; these traits are related to the way posts are formulated and provide subtle context as to what type of information might be missing

- models classify the posts based also on the topic similarity; Figure 3 shows how clickbait posts addressing *savings, retirement* are most likely to be spoiled by multiple phrases or sentences; we use BERT-topic (Grootendorst, 2022) for our analysis

6 Conclusion

Our investigations reveal that transformer models have a limited capacity at identifying the clickbait spoiler types, reaching around 70% accuracy. This raises the question of whether the typology itself is consistent. The majority of confusions occur in multipart examples, as this class is characterized by the presence of discontinuous passages and phrases in the linked document. Upon examining the examples, we found that a significant portion of multipart examples contain very short phrases or passages, making it difficult even for humans to distinguish this category from the others without knowing the actual spoiler.

Lastly, we do not believe this to be the biggest obstacle in reaching the final goal: to diminish the amount of sensationalistic content on social media. After all, this classifier would only feed the decisions of text generator at the following steps. In the end, we believe that content creators and companies will be less enthusiastic to reduce their clickthrough rate of their social-media posts due to such spoilers and possibly new types of language mechanisms that include both images and texts will be used to both bypass the clickbait detectors and exploit the human psychology. In the face of such challenges, we require further research to prepare multilingual and lightweight models that can be easily adapted to a wider variety of media.

References

Jonas Nygaard Blom and Kenneth Reinecke Hansen. 2015. Click bait: Forward-reference as lure in online news headlines. *Journal of Pragmatics*, 76:87–100.

Davide Chicco and Giuseppe Jurman. 2020. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21:1–13.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ullrich KH Ecker, Stephan Lewandowsky, Ee Pin Chang, and Rekha Pillai. 2014. The effects of subtle misinformation in news headlines. *Journal of experimental psychology: applied*, 20(4):323.

Maik Fröbe, Tim Gollub, Matthias Hagen, and Martin Potthast. 2023a. SemEval-2023 Task 5: Clickbait Spoiling. In *17th International Workshop on Semantic Evaluation (SemEval-2023)*.

Maik Fröbe, Matti Wiegmann, Nikolay Kolyada, Bastian Grahm, Theresa Elstner, Frank Loebe, Matthias Hagen, Benno Stein, and Martin Potthast. 2023b. Continuous Integration for Reproducible Shared Tasks with TIRA.io. In *Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023)*, Lecture Notes in Computer Science, Berlin Heidelberg New York. Springer.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Matthias Hagen, Maik Fröbe, Artur Jurk, and Martin Potthast. 2022. Clickbait Spoiling via Question Answering and Passage Retrieval. In *60th Annual Meeting of the Association for Computational Linguistics (ACL 2022)*, pages 7025–7036. Association for Computational Linguistics.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Juwon Hwang, Porismita Borah, Dhavan Shah, and Markus Brauer. 2021. The relationship among covid-19 information seeking, news media use, and emotional distress at the onset of the pandemic. *International Journal of Environmental Research and Public Health*, 18(24).

Vijayasaradhi Indurthi, Bakhtiyar Syed, Manish Gupta, and Vasudeva Varma. 2020. Predicting clickbait strength in online social media. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4835–4846, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Oliver Johnson, Beicheng Lou, Janet Zhong, and Andrey Kurenkov. 2022. Saved you A click: Automatically answering clickbait titles. *CoRR*, abs/2212.08196.

Kenji Kobayashi and Ming Hsu. 2019. Common neural code for reward and information value. *Proceedings of the National Academy of Sciences*, 116(26):13061–13066.

Christina Leslie, Eleazar Eskin, and William Stafford Noble. 2001. The spectrum kernel: A string kernel for svm protein classification. In *Biocomputing 2002*, pages 564–575. World Scientific.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

George Loewenstein. 1994. The psychology of curiosity: A review and reinterpretation. *Psychological bulletin*, 116(1):75.

Peya Mowar, Mini Jain, Ruchika Goel, and Dinesh Kumar Vishwakarma. 2021. Clickbait in youtube prevention, detection and analysis of the bait using ensemble learning. *arXiv preprint arXiv:2112.08611*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Evan Polman, Rachel L. Ruttan, and Joann Peck. 2022. Using curiosity to incentivize the choice of "should" options. *Organizational Behavior and Human Decision Processes*, 173:104192.

Martin Potthast, Tim Gollub, Matthias Hagen, and Benno Stein. The clickbait challenge 2017: Towards a regression model for clickbait strength.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait detection. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016, Padua, Italy,* *March 20–23, 2016. Proceedings 38*, pages 810–817. Springer.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Phillip Rust, Jonas F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. 2023. Language modelling with pixels. In *The Eleventh International Conference on Learning Representations*.

John Shawe-Taylor, Nello Cristianini, et al. 2004. *Kernel methods for pattern analysis*. Cambridge university press.

Vladimir Vapnik. 1999. *The nature of statistical learning theory*. Springer science & business media.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Bram Vijgen et al. 2014. The listicle: An exploring research on an interesting shareable new media phenomenon. *Studia Universitatis Babes-Bolyai-Ephemerides*, 59(1):103–122.

A Appendix

A topic modelling analysis using BERT-topic library (Grootendorst, 2022) of the most common topics per each class shows how models perceive the distribution of different types content in each linked document. Figure 3 shows the topic distributions and Figure 2 shows the keywords in each topic. It is evident that different topic distributions guide the three classes. The first three topics related to Donald Trump, restaurants and fashion have similar distributions in all the three classes.

The main differences are in the class containing *multipart* spoilers, which is characterized by topics defined by keywords such as: *savings, retirement*, *health tips*. And a complete lack of topic 11 on *pets, dogs* and *breeds*.

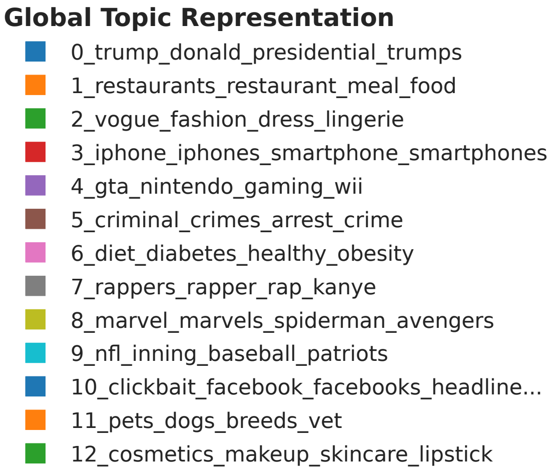

Figure 2: Topic color code, number, and representative keywords.

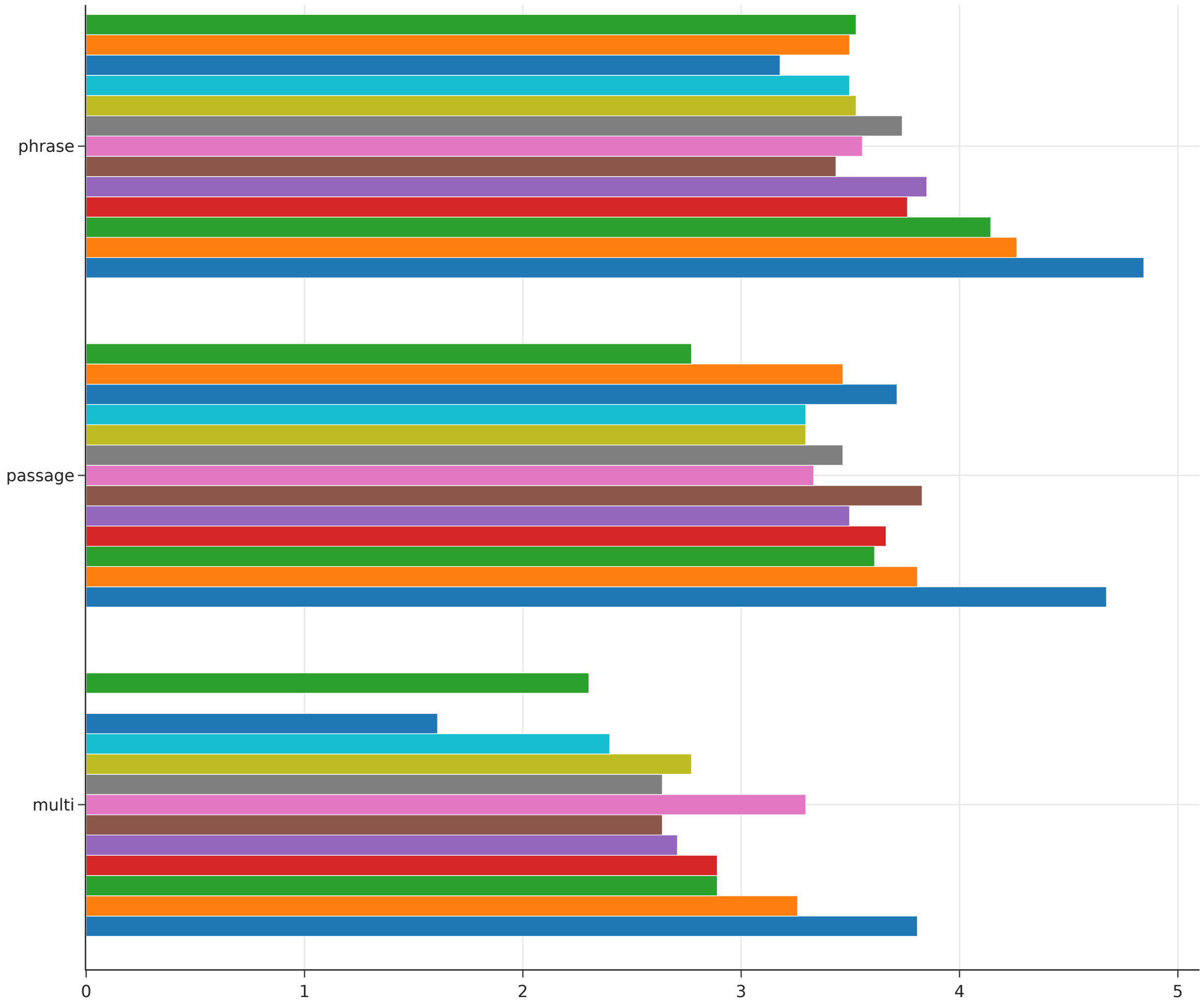

Figure 3: Distribution of different topics for each class.