MeaningCloud at TASS 2018: News Headlines Categorization for Brand Safety Assessment

MeaningCloud en TASS 2018: Clasificación de Titulares de Noticias para Evaluar la Seguridad de Marca

Javier Herrera-Planells, Julio Villena-Román
MeaningCloud LLC
{jherrera, jvillena}@meaningcloud.com

Abstract: This paper describes the participation of MeaningCloud in Task 4 at TASS 2018 (Martínez-Cámara et al., 2018), which is focused on Brand Safety assessment. The objective of systems is to predict whether ads should be hidden for specific news articles, depending on the topics covered and potential negative emotions that could be triggered. Based on the output of our APIs for lemmatization, topics extraction and sentiment analytics, different Natural Language Understanding techniques combined with Machine Learning were tested in our experiments. The experiment that achieved the best result consisted of a Deep Learning algorithm based on Word Embeddings and CNN, trained with features based on the headlines, plus entity extraction, and topic and sentiment analysis.
Keywords: Brand Safety, Unsafe News, Natural Language Understanding, Machine Learning, Feature Selection, Deep Learning, MeaningCloud.

Resumen: Este artículo describe la participación de MeaningCloud en la Tarea 4 de TASS 2018 (Martínez-Cámara et al., 2018), que se centra en la evaluación de la seguridad de marca. El objetivo de los sistemas es predecir si los anuncios deberían ocultarse para artículos de noticias específicos, dependiendo de los temas tratados y de las posibles emociones negativas que pudieran desencadenarse. Utilizando la salida de nuestras APIs para extracción de entidades, lematización, clasificación temática y análisis de sentimiento, nuestro enfoque se basó en probar diferentes técnicas de comprensión del lenguaje natural combinadas con aprendizaje automático en diferentes experimentos. El experimento que alcanzó el mejor resultado ha consistido en un algoritmo de Deep Learning basado en Word Embeddings más CNN, entrenado con características basadas en el texto de los titulares, extracción de entidades, análisis temático y de sentimiento.
Palabras clave: Seguridad de Marca, Noticias seguras e inseguridad, lenguaje natural, aprendizaje automático, selección de características, Deep Learning, MeaningCloud.

1 *Introduction*

In the online advertising context, Brand Safety refers to practices and tools allowing to ensure that an ad will not appear in a context that could affect negatively or directly damage the advertiser's brand.

An article may be considered unsafe for advertising if it triggers negative feelings in the reader. The creation of a system that detects these cases faces some challenges. First, different feelings might be triggered in each reader, depending on their view on topics like religion, economy, or sports. In addition, combinations of pseudo-thematic classifications and sentiment analysis are involved. For example, a reduction of traffic accidents implies a negative feeling because of the mention to car accidents, but the reduction in number actually represents good news.

This paper describes the participation of MeaningCloud in the Task 4 Good Or Bad News of TASS 2018 workshop (Martínez-Cámara et al., 2018), where prediction models have been built for the categorization of news articles headlines into two categories (Safe and Unsafe). In this task, lexical diversity among Spanish and Latin American newspapers is also considered.

ISSN 1613-0073

Two subtasks had to be fulfilled for this Task 4. The first subtask required and evaluated a training algorithm which was fed with a corpus of 1500 headlines in Spanish from various countries. It was afterwards tested against two sets of 500 and 15000 headlines in Spanish from various countries too. The second subtask evaluated the generalization capacity of the algorithm between Spanish from Spain and Spanish from diverse American countries: the training was fed with 250 headlines from newspapers of Spain and tested against 400 headlines from newspapers of Latin America.

The tagged corpus provided was quite well balanced between training, development and test sets with respect to country representation (number of instances), although slightly unbalanced with a higher number of samples in the Unsafe category (64%).

2 *Our Approach*

Our approach is composed by two steps: first, multiple features are extracted from each headline. Then, each feature vector is fed to a machine learning model which finally produces the Safe/Unsafe prediction.

2.1 Feature Generation

Features are extracted using the following public APIs in our text analytics platform: entity extraction and anonymization, lemmatization, and sentiment and topic detection,

2.1.1 Entity Extraction and Anonymization

Entities are detected using the MeaningCloud topics extraction API. This service has been used with raw headlines as the following one:

> *Vídeo muestra cómo Daesh mata a 4 soldados en Níger.*

For this headline, the entities *Daesh* and *Níger* are detected along with their classes: *Organization>TerroristOrganization* and *Location>Country*.

With this information, anonymized versions of the headlines are generated to abstract from references to actual entities that could bias the analysis. Numbers are masked too. For instance:

> *Vídeo muestra cómo #TerroristOrganization# mata a 0 soldados en #Location#.*

In addition to this anonymized version of the headline, the training algorithms were also fed, separately, with the detected entities (*Daesh* and *Níger*).

2.1.2 Text Lemmatization

Headlines are lemmatized after the previous anonymization. The MeaningCloud lemmatization, PoS and parsing API has been used for this task. For the previous example, the lemmatized form is:

> *vídeo mostrar cómo terroristorganization matar a 0 soldado en location.*

2.1.3 Sentiment Detection

Sentiment is detected using the MeaningCloud sentiment analysis API. For instance, given the following raw headline:

> *Animales mueren en zoológico de Venezuela por falta de comida.*

For this headline, a global sentiment N+ is detected. This score belongs to a scale [P+, P, Neutral, N, N+] grading from positive to negative, which we map to a range 0 (P+) to 4 (N+).

The API also provides token-level sentiment (*morir* with sentiment N+, *por falta de comida* with a sentiment N) and subjectivity data (in this case, OBJECTIVE), features which have been omitted in this task.

2.1.4 Topic Detection

News topics (thematic categorization) can be detected either with the MeaningCloud text classification API, using one of the predefined models such as IPTC for news categorization or IAB for advertising market, or aggregating the thematic information returned by the topics extraction API for each detected entity. For this task, we used this second approach. For instance, for the following raw headline:

> *En plena distensión por los Juegos Olímpicos, Kim Jong-Un invitó al presidente de Corea del Sur a Pyongyang.*

The topics extraction API detects four entities: *Juegos Olímpicos* (Event), *Kim Jong-un* (Person), *Corea del Sur* and *Pyongyang* (both Location). Two of them provide thematic information: *Juegos Olímpicos* belongs to *sports* and *Kim Jong-un* belongs to *politics*. So finally *sports* and *politics* are selected as topics.

2.2 Classifiers for Monolingual Classification (subtask 1)

2.2.1 Run 1: Machine Learning

The starting point of our first experiment was a training set of headlines that were anonymized and lemmatized.

We performed an iteration over all of them, creating a list of n-grams that are frequently found in the Unsafe category. Each n-gram was assigned with a higher score if it appeared more frequently in Unsafe than in Safe headlines. Some of the top-ranked Unsafe n-grams in the list after the training process were *morir*, *denunciar*, *asesinar*, *caso de corrupción* and the placeholder for the anonymized entity *terroristorganization*. These n-grams are then used to generate the features for each headline.

One headline is represented by the following feature vector:

- The sum of the scores of Unsafe unigrams found in the headline, weighted to the length of the headline in words.
- Scores for Unsafe bigrams, trigrams and 4-grams separately, in the same way as unigrams.
- The sentiment score (0 to 4), extracted as described in section 2.1.3.
- Features for each of the most frequent topics, such as *sports*, *politics* or *religion*, extracted as described in section 2.1.4.

The most informative features, as shown by the Extra-Trees algorithm (Geurts et al., 2006), were the following: unigram scores (43%), bigram scores (21%), sentiment scores (19%), trigram scores (6%), 4-gram scores (2%), *politics* topics (2%), *football* topics (1%) and *economy* topics (1%).

Then, several machine learning algorithms have been tested for making predictions on these feature vectors, including: KNN (Altman, 1992), random forests (Breiman, 2001), multilayer perceptron, logistic regression, SVM (Vapnik et al., 1995), XGBoost (Chen et al., 2016), and AdaBoost (Freund et al., 2003).

The accuracy of the different algorithms was evaluated using development set. XGBoost was finally chosen as the top-performant algorithm for this experiment. The final settings were a tree booster, learning rate of 0.1, minimum child weight of 1 and maximum depth of 3.

The resulting experiment for this task, trained with 1500 samples and tested with 500 samples (L1 corpus), had the following performance: 71.4% accuracy, 71.7% Macro-F1, 71.3% Macro-Precision, and 72.2% Macro-Recall.

The confusion matrix is shown in Table 1. As it can be observed, the main reason for the errors is the incorrect prediction of Unsafe news as Safe, accounting for 19% of total errors and 32% of the errors in the Unsafe category.

Actual	Predicted	Count
Safe	Safe	153 (30% all, 76% Safe)
Safe	Unsafe	48 (10% all, 24% Safe)
Unsafe	Unsafe	204 (41% all, 68% Unsafe)
Unsafe	Safe	95 (19% all, 32% Unsafe)

Table 1: Run 1, L1 corpus confusion matrix

2.2.2 Run 2: Extended Features

This experiment follows the same principle as the first one: we generate a n-gram score list in our training step, and feature vectors use these scores along the sentiment and topic statistics.

This second experiment extends the n-gram score features by using extra n-gram lists. These additional lists are generated using the non-anonymized version of the headlines. Some of the top scoring resulting n-grams are *FARC*, *Jones Huala* or *caso de Edu Saettone*.

Using this information, which is derived from non-anonymized entities, becomes helpful when categorizing headlines within a similar period.

This approach resulted in an increase of performance over the previous experiment: 73.2% accuracy, 72.5% Macro-F1, 72.3% Macro-Precision, and 72.7% Macro-Recall.

The confusion matrix is shown in Table 2. The detection of Unsafe category has noticeably improved, though the accuracy of Safe category has decreased.

Actual	Predicted	Count
Safe	Safe	141 (28% all, 70% Safe)
Safe	Unsafe	60 (12% all, 30% Safe)
Unsafe	Unsafe	225 (45% all, 75% Unsafe)
Unsafe	Safe	74 (15% all, 25% Unsafe)

Table 2: Run 2, L1 corpus confusion matrix

2.2.3 Run 3: Deep Learning

This experiment, opposite to the previous ones, feeds the machine learning algorithm with a set of words/tokens. A deep learning model based on word embeddings and a convolutional neural network is then used for making predictions.

The following headline will be used for describing the process used in this experiment:

> *Al menos 25 civiles muertos deja ataque contra el Daesh en Siria.*

The same features as described in the previous experiment are generated: preprocessed text, sentiment, topics and entities. All of them are encoded as a set of tokens:

> *al menos 0 civil muerto dejar ataque contra el terroristorganization en location entdaesh entsiria sentiment3 topicpolitics*

The information contained in these tokens is the following:

- Anonymized and lemmatized text: *al menos 0 civil muerto dejar ataque contra el terroristorganization en location.*
- Non anonymized entities: *entdaesh*, *entsiria*.
- Sentiment: *sentiment3*, meaning a sentiment with score 3 (negative, N).
- Topics found: *topicpolitics*.

Afterwards, a deep learning model developed using the Keras framework (Chollet et al., 2015) is trained on this set of tokens. The implementation of the model for this experiment has the following settings:

- Input sequences with length of 23 words (two times 11.5, the average word length), padded for shorter texts with PAD placeholders at the end.
- Embedding generation: 300-dimensional embeddings for the 2241 most frequent words (two thirds of the total 3362 different words). UNK placeholder for words out of selected vocabulary.
- Convolutional neural network, calculating a convolution with 3 different region sizes (Zhang and Wallace, 2015) and 2 filters for each region size. Kernel size of {3,4,5}x300, ReLU activation function (Nair and Hinton, 2010) and max-pooling strategy.
- Two final densely-connected layers with a dropout of 0.25 (Srivastava et al., 2014). The second layer acts as the output layer using a Softmax function.

The resulting model, trained with 1500 samples and tested with 500 samples, showed a high increase in performance: 77.6% accuracy, 76.7% Macro-F1, 76.7% Macro-Precision, and 76.7% Macro-Recall.

Table 3 again shows the confusion matrix.

Actual	Predicted	Count
Safe	Safe	145 (29% all, 72% Safe)
Safe	Unsafe	56 (11% all, 28% Safe)
Unsafe	Unsafe	243 (49% all, 81% Unsafe)
Unsafe	Safe	56 (11% all, 19% Unsafe)

Table 3: Run 3, L1 corpus confusion matrix

There is an improvement in both classes from the previous experiments, most noticeable in the Unsafe category. This confusion matrix is the best among the previous ones if we take into account the risk of considering an Unsafe article as Safe. In this case, ads would be shown by mistake.

2.2.4 Overall Results

Table 4 shows the overall results for subtask 1 of our three experiments, sorted by Macro-F1 which is the comparison metric among participants.

Run Id	Macro-F1
Run 1	0.717
Run 2	0.725
Run 3	0.767

Table 4: Overall results, L1 corpus

Next table 5 shows the final ranking in terms of Macro-F1 for the best run by all participants, sorted by Macro-F1. Our best experiment ranked 4th among 7 participants.

Group	Macro-F1
INGEOTEC	0.795
ELiRF-UPV	0.790
rbnUGR	0.774
MEANINGCLOUD	0.767
SINAI	0.728
lone_wolf	0.700
TNT-UA-WFU	0.492

Table 5: Subtask 1, L1 corpus team ranking

Finally, the results over the L2 corpus (including 13 152 headlines), tagged by pooling submissions and based on the vote of majority, are shown in Table 6. Our best experiment ranked 4th again among all participants. The improvement of results with respect to the other corpus (L1) may be not real because of the pooling (the decision of the majority may be wrong anyway).

Group	Macro-F1
ELiRF-UPV	0.883
rbnUGR	0.873
INGEOTEC	0.866
MEANINGCLOUD	0.793
SINAI	0.773
TNT-UA-WFU	0.544

Table 6: Subtask 1, L2 corpus team ranking

2.3 Classifiers for Multilingual Classification (subtask 2)

Run 3 was, apparently, the top performant among the other experiments in subtask 1, so it was selected for subtask 2. The model was trained with 250 headlines from newspapers of Spain, and tested against 408 headlines from newspapers of America.

The results were the following: 65.8% accuracy, 65.1% Macro-F1, 64.7% Macro-Precision, and 65.4% Macro-Recall.

The confusion matrix is shown in Table 7. Obviously, results are considerably worse than in the first task, as the information available for training is extremely reduced.

Actual	Predicted	Count
Safe	Safe	99 (24% all, 63% Safe)
Safe	Unsafe	57 (14% all, 37% Safe)
Unsafe	Unsafe	169 (42% all, 67% Unsafe)
Unsafe	Safe	84 (20% all, 33% Unsafe)

Table 7: Run 3 subtask 2 confusion matrix

Finally, Table 8 shows the ranking in terms of Macro-F1 in subtask 2 for the best run by all participants, sorted by Macro-F1. Our best experiment ranked 4th among 5 participants.

Group	Macro-F1
INGEOTEC	0.719
ELiRF-UPV	0.699
rbnUGR	0.683
MEANINGCLOUD	0.651
ITAINNOVA	0.617

Table 8: Subtask 2 team ranking

Results in this subtask are, as expected, lower than for subtask 1, for all teams. The ranking among teams stays the same, so, apparently, the lack of information (or the lack of generalization of the models) affects the same to all groups.

3 *Conclusions*

In this paper we described a system for detecting headlines of news articles that might be unsafe for advertising. We have incorporated different preprocessing techniques, such as text lemmatization, entity extraction and anonymization, topic detection and sentiment analysis.

Then, we have evaluated several classification algorithms, from n-gram scoring to embeddings and deep learning models.

Three techniques were found to significantly improve the accuracy of the model: providing both the anonymized text and the non-anonymized entities separately, include the sentiment pre-detection and the use of a deep learning approach for the model training.

Disclaimer

MeaningCloud is one of the co-organizers of TASS since the first edition in 2012, and, specifically this year, of Task 4 Good Or Bad News. Our participation in this task has been completely blind, without making use of any information or dataset not provided to the rest of the participants.

We are also sponsoring TASS 2018 with prizes for the best teams. Obviously, as insiders, we were never eligible for the prize, should our experiments had been the top-performant.

References

Altman, N. S. 1992. An Introduction to Kernel and Nearest-Neighbor Non-Parametric Regression. *The American Statistician*, 46(3), 175-185.

Breiman, L. 2001. Random Forests. *Machine learning*, 45(1), 5-32.

Chen, T., and C. Guestrin. 2016. Xgboost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794). ACM.

Chollet, F. 2015. Keras. GitHub. https://github.com/keras-team/keras

Cortes, C., and V. Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3), 273-297.

Freund, Y., R. Iyer, R.E. Schapire, and Y. Singer. 2003. An Efficient Boosting Algorithm for Combining Preferences. *The*

Journal of Machine Learning Research, 4 (Nov), 933-969.

Geurts, P., D. Ernst, and L. Wehenkel. 2006. Extremely Randomized Trees. *Machine learning*, 63(1), 3-42.

Martínez-Cámara, E., Y. Almeida-Cruz, M.C. Díaz-Galiano, S. Estévez-Velarde, M.A. García-Cumbreras, M. García-Vega, Y. Gutiérrez, A. Montejo Ráez, A. Montoyo, R. Muñoz, A. Piad-Morffis, and J. Villena-Román. 2018. Overview of TASS 2018: Opinions, Health and Emotions. In *Proceedings of TASS 2018: Workshop on Semantic Analysis at SEPLN (TASS 2018)*. CEUR Workshop Proceedings, vol 2172, Sevilla, Spain, September 2018. CEUR-WS.

Nair, V., and G. E. Hinton. 2010. Rectified Linear Units improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on Machine Learning* (ICML-10) (pp. 807-814).

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1), 1929-1958.

Zhang, Y., and B. Wallace. 2015. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. CoRR 2015.