TweetNLP: Cutting-Edge Natural Language Processing for Social Media

Jose Camacho-Collados[1]   **Kiamehr Rezaee**[1]   **Talayeh Riahi**[1]   **Asahi Ushio**[1]
Daniel Loureiro[1]   **Dimosthenis Antypas**[1]   **Joanne Boisson**[1,6]   **Luis Espinosa-Anke**[1,6]
Fangyu Liu[2]   **Eugenio Martínez-Cámara**[3]   **Gonzalo Medina**[3]
Thomas Buhrmann[4]   **Leonardo Neves**[5]   **Francesco Barbieri**[5]

[1]Cardiff NLP, Cardiff University, UK   [2]LTL, University of Cambridge, UK
[3]DaSCI, University of Granada, Spain   [4]Graphext, Spain   [5]Snap Inc., USA   [6]AMPLYFI, UK
`cardiffnlp.contact@gmail.com`

Abstract

In this paper we present TweetNLP, an integrated platform for Natural Language Processing (NLP) in social media. TweetNLP supports a diverse set of NLP tasks, including generic focus areas such as sentiment analysis and named entity recognition, as well as social media-specific tasks such as emoji prediction and offensive language identification. Task-specific systems are powered by reasonably-sized Transformer-based language models specialized on social media text (in particular, Twitter) which can be run without the need for dedicated hardware or cloud services. The main contributions of TweetNLP are: (1) an integrated Python library for a modern toolkit supporting social media analysis using our various task-specific models adapted to the social domain; (2) an interactive online demo for codeless experimentation using our models; and (3) a tutorial covering a wide variety of typical social media applications.

1 Introduction

Today's society cannot be understood without the role of social media. Online users connect more and more via platforms that enable content sharing, either generic or around specific topics, and do this by means of text-only messages, or augmenting them with multimedia content such as pictures, audio or video. As such, these platforms have been used to understand user, group and organization-wide behaviours (Yang et al., 2021; Hu et al., 2021). In particular, Twitter, which is the main platform studied in this paper, has long been an important resource for understanding society at large (Weller et al., 2013). Twitter is interesting for NLP because it embodies many features that are natural in spontaneous and ever-evolving fast-paced communication. However, the majority of NLP research focuses on optimizing model development against training data and evaluation benchmarks which are, at worst, reasonably clean (e.g., news articles, blog posts or Wikipedia). Consequently, when deployed *in the wild*, features such as noisiness, multilinguality, immediacy, slang, technical jargon, lack of context, platform-specific restrictions on message length, emoji and other modalities, etc. become core communicative variables that need to be factored in. Indeed, even traditional NLP tasks such as normalization (Han and Baldwin, 2011; Baldwin et al., 2015), POS tagging (Derczynski et al., 2013), sentiment analysis (Poria et al., 2020) or named entity recognition (Ritter et al., 2011; Baldwin et al., 2013) have been shown to produce suboptimal results in the context of social media.

Given the above, we put forward TweetNLP (`tweetnlp.org`), which offers a full-fledged NLP platform specialized in Twitter. The backbone of TweetNLP consists of Transformer-based language models that have been trained on Twitter (Barbieri et al., 2020, 2022; Loureiro et al., 2022). Then, these specialized language models have been further fine-tuned for specific NLP tasks on Twitter data. These models have already proved highly popular, with thousands of downloads from the Hugging Face model hub every month (Wolf et al., 2020).[1] TweetNLP integrates all these resources into a single platform. With a simple Python API, TweetNLP offers an easy-to-use way to leverage cutting-edge NLP models in a variety of social media tasks. Despite the trend of ever-larger language models (Shoeybi et al., 2019; Brown et al., 2020), TweetNLP is more focused on the general user and applicability, and therefore integrates base models which are easily run in standard computers or on free cloud services. Finally, all models can be accessed from an interactive online demo, offering anyone the possibility to test models and perform

---

[1]Most notably, the sentiment analysis model has been the most downloaded model in the Hugging Face model hub in January 2022, with over 15M downloads. Similarly, the Tweet-Eval benchmark, in which most task-specific Twitter models are fine-tuned, has been the second most downloaded dataset in April 2022, with over 150K downloads.

real-time analysis on Twitter.

2 Related Work

General-purpose NLP libraries have been available for many years. Starting from the Java-based CoreNLP (Manning et al., 2014) to the more recent Python-based library Stanza (Qi et al., 2020). More recently, libraries such as spaCy[2] have been ubiquitous in NLP, both in research and industry. Finally, in the language models and Transformers era, the Hugging Face Transformer hub has become indispensable for state-of-the-art NLP (Wolf et al., 2020), which is also leveraged for our library TweetNLP. However, none of these libraries is specialized in social media or Twitter.

As for libraries developed specifically for social media, these are more limited and mostly associated with low-level tasks such as tokenization, part-of-speech (Owoputi et al., 2013) tagging or dependency parsing (Kong et al., 2014), and initially available for Java. The most recent Twitter-specific Python library is TweebankNLP (Jiang et al., 2022) based on Stanza. This library provides state-of-the-art models on tokenization and lemmatization, besides competitive models on NER, part-of-speech tagging and dependency parsing. In contrast, TweetNLP focuses on specialized Twitter-specific language models for downstream tasks such as sentiment analysis and offensive language identification.

3 Models and Functionalities

TweetNLP is versatile in that it covers a wide range of tasks and applications. The backbone of TweetNLP are transformer-based language models, which are covered in Section 3.1. The concrete NLP tasks integrated in TweetNLP are presented in Section 3.2. Finally, in Section 3.3 we present embeddings used to represent words and tweets. All TweetNLP model checkpoints are available in the appendix.

3.1 Language models

Language models are at the core of TweetNLP. Instead of relying on general-purpose models (Devlin et al., 2019) or training a language model from scratch (Nguyen et al., 2020), we start from RoBERTa (Liu et al., 2019) and XLM-R (Conneau et al., 2020) checkpoints and continue pre-training on Twitter-specific corpora. This was shown to be

---

[2]https://spacy.io

generally more reliable for the amount of text analysed in Barbieri et al. (2020). Given our aim for democratizing the usage of specialized language models for social media, another important feature of TweetNLP is the relatively small size of the language models. To this end, all language models rely on the equivalent of a RoBERTa-base or XLM-R-base architecture. These models are efficient on standard hardware and free-tiers of cloud computing services, with reasonable speed even without GPU support.

TweetEval (Barbieri et al., 2020). This model was initially released as part of the TweetEval project. It is based on a RoBERTa-base architecture using the original model as an initial check point (Liu et al., 2019). Later, this language model was further pre-trained on a corpus of 60M tweets from May 2018 to August 2019.

TimeLMs (Loureiro et al., 2022). This model is initially trained on the same Twitter corpus used by Barbieri et al. (2020). The main difference lies on a few preprocessing improvements applied to the underlying corpus, including measures to reduce potential spam and near duplicates, and more recent corpora used for continual pretraining. The overall quantity of tweets is therefore larger, as the model is regularly updated (every 3 months) with a fixed number of additional tweets. The most recently released TimeLMs model to date is pre-trained on 132M tweets until the end of June 2022.

XLM-T (Barbieri et al., 2022). This model was trained on 198M tweets on over thirty languages from May 2018 to March 2020, following a similar strategy to Barbieri et al. (2020). In this case, the initial checkpoint was XLM-R-base (Conneau et al., 2020).

3.2 Supported tasks

In the following we describe the tasks supported by TweetNLP. For the tweet classification tasks included in TweetEval, and for topic classification, we simply fine-tune the models described above on the corresponding datasets, as described in Barbieri et al. (2020). For model fine-tuning on named entity recognition, we rely on the T-NER library (Ushio and Camacho-Collados, 2021), which is also integrated into TweetNLP.

Sentiment Analysis. The sentiment analysis task integrated in TweetNLP consists of predicting the

sentiment of a tweet with one of the three following labels: positive, neutral or negative. The base dataset for English is the unified TweetEval version of the Semeval-2017 dataset from the task on *Sentiment Analysis in Twitter* (Rosenthal et al., 2017). Moreover, for the languages other than English we include the datasets integrated in UMSAB (Barbieri et al., 2022), namely Arabic (Rosenthal et al., 2017), French (Benamara et al., 2017), German (Cieliebak et al., 2017), Hindi (Patra et al., 2015), Italian (Barbieri et al., 2016), Portuguese (Brum and Volpe Nunes, 2018), and Spanish (Díaz-Galiano et al., 2018).

Emotion Recognition. Given a tweet, this task consists of associating it with its most appropriate emotion. As a reference dataset we use the SemEval 2018 task on *Affect in Tweets* (Mohammad et al., 2018), simplified to only the four emotions used in TweetEval: anger, joy, sadness and optimism.

Emoji Prediction. The goal of emoji prediction is to predict the final emoji on a given tweet. The dataset used to fine-tune our models is the TweetEval adaptation from the SemEval 2018 task on *Emoji Prediction* (Barbieri et al., 2018), including 20 emoji as labels.

Irony Detection. This is a binary classification task that aims at detecting whether a tweet is ironic or not. It is based on the *Irony Detection* dataset from the SemEval 2018 task (Van Hee et al., 2018).

Hate Speech Detection. The hate speech dataset consists of detecting whether a tweet is hateful towards women or immigrants. It is based on the *Detection of Hate Speech* task at SemEval 2019 (Basile et al., 2019).

Offensive Language Identification. The task consist of identifying any form of offensive language in a tweet. The dataset is based on the SemEval 2019 task on *Identifying and Categorizing Offensive Language in Social Media* (Zampieri et al., 2019).

Stance Detection. Given a target topic and a tweet, stance detection consists of assessing whether the author of the tweet has a positive, neutral or negative position towards the target. The dataset considered was initially released for the SemEval 2016 task on *Detecting Stance in Tweets* (Mohammad et al., 2016).

Topic Classification. The aim of this task is, given a tweet, assign topics related to its content. The task is formulated as a supervised multi-label classification problem where each tweet is assigned one or more topics from a total of 19 available topics. The topics were carefully curated based on Twitter trends with the aim to be broad and general, consisting of classes such as: *arts and culture*, *music*, or *sports*. The underlying tweet topic classification dataset contains over 10K manually-labeled tweets (Antypas et al., 2022).

Named Entity Recognition. The goal of named entity recognition (NER) is to find entities and identify their entity types in a given sentence. The underlying Twitter NER dataset is composed of over 10K tweets which were annotated (internally) with seven entity types.[3]

3.3 Embeddings

In addition to the language models and their supported tasks, we also release high-quality vector representation models for words and tweets, i.e., *embeddings* (Pilehvar and Camacho-Collados, 2020). These relatively low-dimensional vector representations can be exploited for a different range of applications and analyses such as word/tweet similarity or tweet retrieval, to name a few.

Word embeddings. TweetNLP word embeddings are based on fastText (Bojanowski et al., 2017) and trained on the same corpora used to train the language models described in Section 3.1. In particular, we trained two sets of embeddings: (1) a monolingual English model trained with the TimeLMs Twitter corpus until the end of 2021; and (2) a multilingual model trained with the Twitter corpus used for XLM-T. Both models were trained using the official fastText package with 300 dimensions, minimum n-gram size 2, maximum n-gram size 12, and remaining parameters set to defaults.

Tweet embeddings. For tweet embeddings, we pulled tweet-reply pairs from the Twitter API and trained contrastive embeddings with an InfoNCE loss (Oord et al., 2018). For tweets with multiple replies, we randomly sampled one reply. In training, one mini-batch is composed of a list of tweet-reply pairs. The tweet-reply pairs are regarded as

---

[3]More details about the datasets for topic classification and named entity recognition will be provided at a later stage. Datasets were annotated internally in Snap and we are working on releasing them to the public according to regulations.

positive samples; the enumeration of all other possible combination of tweet-reply, tweet-tweet, and tweet-reply pairs are regarded as negative samples. The contrastive InfoNCE loss then pulls positive pair representations close while pushes negative representations away from each other. Training was performed on 1.1M tweet-reply pairs, and we collected a separate tweet-reply set of 10k pairs for selecting the model checkpoint.

4 TweetNLP Python library

The TweetNLP Python library has been integrated into pypi[4] and therefore is easily accessible and can be installed from pip ("pip install tweetnlp"). All the details on how to use TweetNLP are in the associated Github repository, which is released fully open-source: https://github.com/cardiffnlp/tweetnlp.

Once installed, loading and using a fine-tuned model on any specific task can be done as follows.

```python
from tweetnlp import load
tweet = "I love Paris!!"
# Sentiment Analysis
model = load('sentiment')
model.sentiment(tweet)
# Tweet Embeddings
model = load('sentence_embedding')
model.embedding(tweet)
# Masked Language Model
model = load('language_model')
tweet = "I love <mask>!!"
model.mask_prediction(tweet)
```

With the *load* statement, the associated fine-tuned language models are loaded in the background. Users can then get the predictions for any given sentence or tweet with a simple pre-defined function (e.g., *.sentiment* or *.predict*). Custom loading of existing fine-tuned language models not included in TweetNLP is also possible. The same functionalities apply to all the other tasks described in Section 3.2.

5 Tutorials

In addition to the Python library presented in the previous section, TweetNLP offers access to the underlying Python code structured in instructive Google Colab notebooks with starter code and examples (https://tweetnlp.org/get-started/). These notebooks are aimed at users with varying degrees of experience in NLP and social media processing. In the following we list the currently existing tutorials and a brief description:

Introduction to TweetNLP. In this initial introduction, users learn how to use the TweetNLP Python library to make use of specialized models in social media for a wide variety of tasks from sentiment analysis to named entity recognition.

Getting data from Twitter. This notebook helps users understand the Twitter API[5] and how to interact with it. More importantly, there are concrete examples on how to retrieve data (i.e. tweets) from Twitter, usually given a hashtag or a keyword.

Custom fine-tuning. In this notebook users can learn to fine-tune any given language model on a specific task (e.g. sentiment analysis). For this, we will take advantage of the TweetEval task data and unified format (Barbieri et al., 2020). Additionally, users can learn how to easily evaluate language models on TweetEval.

Word embeddings. With this notebook users can learn how to train their own word embeddings on custom data using Gensim[6] (Řehůřek and Sojka, 2010). The notebook also includes examples on how to get similarity scores from Twitter-specific word embeddings, or how to obtain the nearest neighbour words from a given input word.

Language models over time. This notebook leverages the TimeLMs library (Loureiro et al., 2022). Users can learn how to make use of language models that have been trained in short periods of time since 2019 until recently.

Tweet embeddings. This notebook contains examples on how to transform a tweet into a vector (embedding) and how these enable important applications such as tweet similarity and retrieval.

6 Demo

In addition to the Python-based library and tutorials, we developed a comprehensive web-based demo integrating all our models, available at https://tweetnlp.org/demo/. The goal of the demo is for any user to be able to test our models and get predictions. In particular, the model includes the following five functionalities:

---

[4] https://pypi.org/project/tweetnlp/

[5] https://developer.twitter.com/en/docs/twitter-api

[6] https://radimrehurek.com/gensim/

Sentence/tweet classification (Figure 1). Users can input a sentence or a tweet (including a tweet URL) and the output is a plot display of the confidence of the model with respect to its predictions. This demo includes all tweet classification tasks supported in English (see Section 3.2), as well as a multilingual sentiment analysis model based on XLM-T.
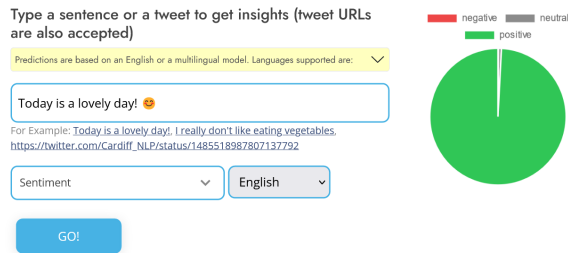

Figure 1: TweetNLP tweet classification demo.

Hashtag analysis (Figure 2). This demo directly interacts with the Twitter API. Users can type a hashtag (or any keyword), initial and end dates, task and language. The system will then retrieve tweets for the given time interval and compute an aggregated analysis of the results. Languages supported for this demo are available in the appendix.
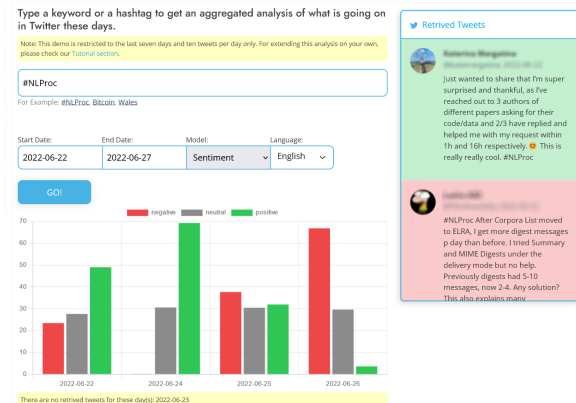

Figure 2: TweetNLP hashtag analysis demo. The output is a bar plot that shows the sentiment of the retrieved tweets over time for the input hashtag #NLProc.

Word prediction (Figure 3). Masked language models utilized in TweetNLP are trained to predict unknown (or *masked*) words within a sentence. For this demo, users can input a sentence with a masked word and the system will show the most likely words as given by the masked language model, in order of confidence.
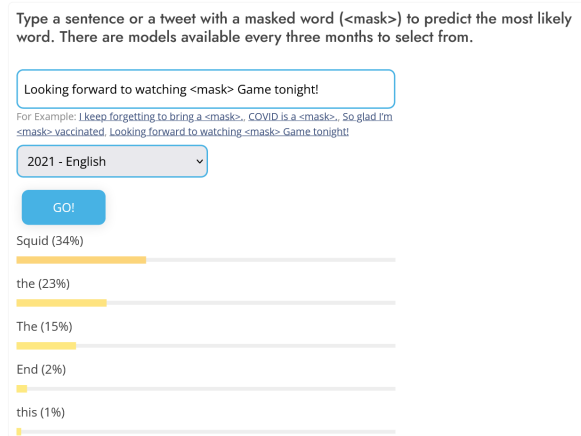

Figure 3: TweetNLP word prediction demo.

Tweet similarity (Figure 4). Given two short pieces of text (e.g., two sentences or two tweets), this demo displays their cosine similarity score on a 0-100 scale as provided by our default tweet embedding model.
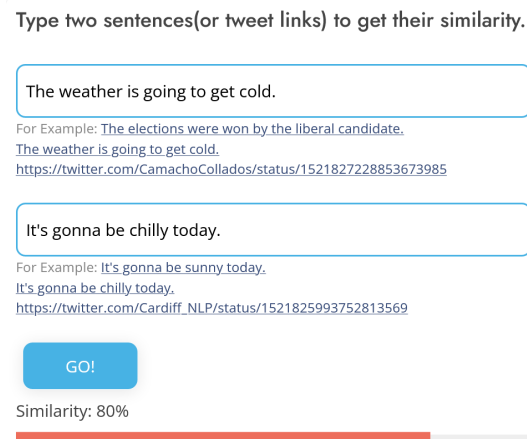

Figure 4: TweetNLP tweet similarity demo.

Named Entity Recognition (Figure 5). Given a tweet or a sentence, this NER demo locates its named entities and infers their types.

7 Evaluation

In this section, we provide experimental results of the default models integrated into TweetNLP.

7.1 Experimental setting

Datasets. For the evaluation we utilized all the train/validation/test splits described in Section 3.2. In particular, we relied on the TweetEval-released datasets for all tweet classification tasks except for topic classification.
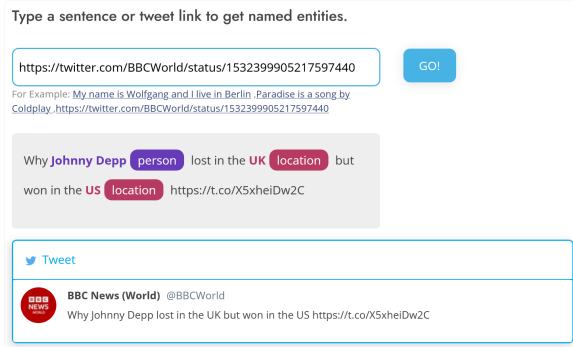
Figure 5: TweetNLP Named Entity Recognition demo.

Default TweetNLP language models. While in TweetNLP all Twitter-specific language models are included, we use as a default (1) TimeLMs trained until December 2021 for English and (2) XLM-T for the languages other than English and multilingual tasks. These models are then fine-tuned to the corresponding tasks as described in Section 3.2.

Comparison systems. We report the performance of all original TweetEval baselines (Barbieri et al., 2020): a frequency-based SVM classifier, fastText (Joulin et al., 2017), a Bidirectional LSTM, RoBERTa-base (Liu et al., 2019), a RoBERTa-base model trained on Twitter from scratch (RoB-Twitter) and the original TweetEval RoBERTa-base model. As another baseline we include BERTweet (Nguyen et al., 2020), trained on almost 1 billion tweets from 2013 to 2019.

Language model fine-tuning. Fine-tuning is performed on the training sets of each corresponding dataset, using their corresponding development sets for validation. We followed TweetEval training protocols for tweet classification, where only the learning rate and number of epochs are tuned (Barbieri et al., 2020). All reported results for language models are based on an average of three runs.

7.2 Results

Table 1 shows the main results of our TweetNLP default language model and comparison systems on nine Twitter-based tasks.[7] The default TimeLMs-21 model achieves the overall results on most tasks, especially comparing it with a comparable general-purpose RoBERTa-based model. In the following we also provide details of our experimental results

on languages other than English , and for the integrated word and tweet embedding models.

Multilingual sentiment analysis results. In addition to the English evaluation, we report results on multilingual sentiment analysis (Table 2). The evaluation is performed on the UMSAB multingual sentiment analysis benchmark (Barbieri et al., 2022). For this evaluation we compare XLM-T fine-tuned on all the language-specific training sets of UMSAB with XLM-R (Conneau et al., 2020) using the same fine-tuning strategy. As an additional indicative baseline, we include fastText trained on the language-specific training sets. As can be observed, our domain-specific XLM-T language model achieves the best overall results in all languages, further reinforcing the importance of in-domain language model training.

Word embedding results. As a sanity check to verify the quality of the word embeddings, we simply test them on standard word similarity datasets: The WS-Sim similarity and WS-Rel relatedness subsets (Agirre et al., 2009) from WordSim-353 (Finkelstein et al., 2002), SemEval-2017 (Camacho-Collados et al., 2017) and MEN (Bruni et al., 2014). Then, we compared the results with the pre-trained fastText model trained on the Common Crawl (Bojanowski et al., 2017), and Wikipedia. According to Spearman correlation, the results of our Twitter embeddings were 0.77 (WS-Sim), 0.72 (WS-Rel), 0.69 (SemEval), and 0.79 (MEN).[8] In contrast, the pre-trained fast-Text Common Crawl results were 0.84 (WS-Sim), 0.64 (WS-Rel), 0.67 (SemEval), and 0.81 (MEN). We should note that these datasets are not specific to social media and even so, our trained embeddings outperform the standard pre-trained fastText in two datasets. In particular, there seems to be a marked difference between similarity and relatedness, where our Twitter embeddings appear to be more suited to relatedness.

Tweet embedding results. For tweet embeddings we explore a tweet retrieval task setting which consists of finding the reply to a given tweet from the 10k replies in the search space. We randomly sampled 3k tweet-reply pairs that do not

---

[7]The BERTweet result on Irony is marked with * as their pre-training corpus overlapped with the Irony dataset, which was constructed using distant supervision.

[8]While not directly comparable given the different sizes, we also compared with our previously-released Twitter-specific 100-dimensional fastText embeddings (Camacho-Collados et al., 2020). The results for these embeddings were consistently lower: 0.65 (WS-Sim), 0.43 (WS-Rel), 0.52 (SemEval), and 0.76 (MEN).

	Emoji	Emotion	Hate	Irony	Offensive	Sentiment	Stance	Topic	NER
SVM	29.3	64.7	36.7	61.7	52.3	62.9	67.3	30.5	-
fastText	25.8	65.2	50.6	63.1	73.4	62.9	65.4	24.0	-
BLSTM	24.7	66.0	52.6	62.8	71.7	58.3	59.4	27.0	-
RoB-Base	30.9	76.1	46.6	59.7	79.5	71.3	68.0	50.1	58.0
RoB-Twitter	29.3	72.0	46.9	**65.4**	77.1	69.1	66.7	-	-
TweetEval	31.4	78.5	52.3	61.7	80.5	72.6	69.3	56.8	56.8
BERTweet	33.4	79.3	**56.4**	*82.1**	79.5	73.4	71.2	52.7	58.7
TweetNLP (TimeLMs-21)	**34.0**	**80.2**	55.1	64.5	**82.2**	**73.7**	**72.9**	**58.8**	**59.7**
Evaluation metric	M-F1	M-F1	M-F1	$F^{(i)}$	M-F1	M-Rec	AVG (F)	M-F1	M-F1

Table 1: Test results in the nine TweetNLP-supported tasks.

	Arabic	English	French	German	Hindi	Italian	Portuguese	Spanish	ALL
fastText	45.98	50.85	54.82	59.56	37.08	54.65	55.05	50.06	51.01
XLM-R	64.31	68.52	70.52	72.84	53.39	68.62	69.79	66.03	66.75
TweetNLP (XLM-T)	**66.89**	**70.63**	**71.18**	**77.35**	**56.35**	**69.06**	**75.42**	**68.52**	**67.91**

Table 2: Sentiment analysis results (Macro-F1) on the UMSAB unified benchmark. XLM-R and TweetNLP models are fine-tuned on the training sets of all languages.

overlap with training data and split them into 3 sets of 1k pairs. We report accuracy@1 and average models' performance on the 3 sets. We also include results on sentence similarity, using the STS-benchmark (Cer et al., 2017) and reporting Spearman's correlation. We list tweet-reply retrieval accuracy and STS-benchmark Spearman's correlation in Table 3. We compare with recent supervised (Reimers and Gurevych, 2019, Sentence-BERT; all-mpnet-base-v2), and unsupervised (Liu et al., 2021, Mirror-BERT), (Gao et al., 2021, SimCSE) sentence embedding models.[9] On the task of tweet-reply retrieval, our tweet-embeddings model significantly outperforms all-mpnet-base-v2 trained with around 1B sentence pairs. This highlights the importance of in-domain training. On the STS-Benchmark, all-mpnet-base-v2 achieves the best performance and our tweet-embeddings perform the worst among baselines but they are generally in a similar ballpark. To complement this evaluation, we plan to test our tweet embeddings with a textual similarity dataset in the tweet domain in the future.

Model	Retrieval	STS
Sentence-BERT	6.1	77.0
all-mpnet-base-v2	15.8	**83.4**
Mirror-RoBERTa	8.8	79.6
SimCSE-RoBERTa	9.2	80.3
TweetNLP (Tweet-embeddings)	**26.7**	70.7

Table 3: Results of sentence and tweet embedding models on tweet-reply retrieval and the STS-benchmark.

8 Conclusion and Future Work

In this demo paper we have presented TweetNLP, an all-round platform for NLP specialized in social media. The platform is powered by relatively lightweight language models trained on Twitter, and adapted (fine-tuned) to various popular NLP tasks on social media, such as sentiment analysis and offensive language identification. In addition

___
[9]Baseline checkpoint links are included in the Appendix.

to sharing the models, TweetNLP provides an online demo, a Python library, and a tutorial to make the most of the models, regardless of the expertise of the user. TweetNLP also enables easy inspection of the models by non-programmers, which can help identify harmful biases or errors, that in turn would help improve the models in the future.

While this first release version of TweetNLP is self-contained and complete, our goal is to keep updating it with both new models and tasks. Since social media data is at the core of TweetNLP, we are planning to develop new datasets and models for social media tasks. In particular, our idea is to go beyond tweet classification tasks, which are currently well covered in TweetNLP. For instance, low-level tasks such as syntactic parsing and part-of-speech tagging has been traditionally hard in noisy environments such as social media. Finally, in the future we are also planning to extend TweetNLP to other social media platforms such as Reddit, LinkedIn or Instagram, and to provide support for languages other than English in a wider variety of tasks.

9 Impact Statement

This paper deals with social media data, in particular with Twitter. All Twitter regulations were followed and data was extracted through the official Twitter API. To mitigate the potential effect of working with this type of data, all dataset-related tweets were anonymized, with URLs removed. In most cases dataset creators made an effort to remove offensive or harmful content from the tweets. Nonetheless, models trained on this data may amplify existing biases present in the social media platform. While this is in many cases unavoidable, we hope that by making this demo public with model prototypes, experts will be able to more easily inspect these biases and we will be able to better understand the potential biases of models trained on this type of data.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, Boulder, Colorado. Association for Computational Linguistics.

Dimosthenis Antypas, Asahi Ushio, Jose Camacho-Collados, Leonardo Neves, Vítor Silva, and Francesco Barbieri. 2022. Twitter topic classification. *arXiv preprint arXiv:2209.09824*.

Timothy Baldwin, Paul Cook, Marco Lui, Andrew MacKinlay, and Li Wang. 2013. How noisy social media text, how diffrnt social media sources? In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 356–364, Nagoya, Japan. Asian Federation of Natural Language Processing.

Timothy Baldwin, Marie Catherine de Marneffe, Bo Han, Young-Bum Kim, Alan Ritter, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 126–135, Beijing, China. Association for Computational Linguistics.

Francesco Barbieri, Valerio Basile, Danilo Croce, Malvina Nissim, Nicole Novielli, and Viviana Patti. 2016. Overview of the evalita 2016 sentiment polarity classification task. In *Proceedings of third Italian conference on computational linguistics (CLiC-it 2016) & fifth evaluation campaign of natural language processing and speech tools for Italian. Final Workshop (EVALITA 2016)*.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650, Online. Association for Computational Linguistics.

Francesco Barbieri, Jose Camacho-Collados, Francesco Ronzano, Luis Espinosa-Anke, Miguel Ballesteros, Valerio Basile, Viviana Patti, and Horacio Saggion. 2018. SemEval 2018 task 2: Multilingual emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 24–33, New Orleans, Louisiana. Association for Computational Linguistics.

Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. Xlm-t: Multilingual language models in twitter for sentiment analysis and beyond. In *Proceedings of the Language Resources and Evaluation Conference*, pages 258–266, Marseille, France. European Language Resources Association.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Farah Benamara, Cyril Grouin, Jihen Karoui, Véronique Moriceau, and Isabelle Robba. 2017. Analyse d'opinion et langage figuratif dans des tweets: présentation et résultats du défi fouille de textes deft2017. In *Défi Fouille de Textes DEFT2017. Atelier TALN 2017*. Association pour le Traitement Automatique des Langues (ATALA).

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda

Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Henrico Brum and Maria das Graças Volpe Nunes. 2018. Building a sentiment corpus of tweets in Brazilian Portuguese. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *J. Artif. Intell. Res.(JAIR)*, 49(1-47).

Jose Camacho-Collados, Yerai Doval, Eugenio Martínez-Cámara, Luis Espinosa-Anke, Francesco Barbieri, and Steven Schockaert. 2020. Learning cross-lingual word embeddings from twitter via distant supervision. In *Proceedings of the international AAAI conference on web and social media*, volume 14, pages 72–82.

Jose Camacho-Collados, Mohammad Taher Pilehvar, Nigel Collier, and Roberto Navigli. 2017. SemEval-2017 task 2: Multilingual and cross-lingual semantic word similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 15–26, Vancouver, Canada. Association for Computational Linguistics.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Mark Cieliebak, Jan Milan Deriu, Dominic Egger, and Fatih Uzdilli. 2017. A Twitter corpus and benchmark resources for German sentiment analysis. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 45–51, Valencia, Spain. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Leon Derczynski, Alan Ritter, Sam Clark, and Kalina Bontcheva. 2013. Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 198–206, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Manuel C. Díaz-Galiano, Eugenio Martínez-Cámara, M. Ángel García Cumbreras, Manuel García Vega, and Julio Villena Román. 2018. The democratization of deep learning in tass 2017. *Procesamiento del Lenguaje Natural*, 60:37–44.

Lev Finkelstein, Gabrilovich Evgeniy, Matias Yossi, Rivlin Ehud, Solan Zach, Wolfman Gadi, and Ruppin Eytan. 2002. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 368–378, Portland, Oregon, USA. Association for Computational Linguistics.

Danqi Hu, Charles M Jones, Valerie Zhang, and Xiaoyan Zhang. 2021. The rise of reddit: How social media affects retail investors and short-sellers' roles in price discovery. *Available at SSRN 3807655*.

Hang Jiang, Yining Hua, Doug Beeferman, and Deb Roy. 2022. Annotating the tweebank corpus on named entity recognition and building nlp models for social media analysis. In *Proceedings of the Language Resources and Evaluation Conference*, pages 7199–7208, Marseille, France. European Language Resources Association.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1001–1012, Doha, Qatar. Association for Computational Linguistics.

Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021. Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*.

Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-collados. 2022. TimeLMs: Diachronic language models from Twitter. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 251–260, Dublin, Ireland. Association for Computational Linguistics.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. SemEval-2018 task 1: Affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1–17, New Orleans, Louisiana. Association for Computational Linguistics.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California. Association for Computational Linguistics.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online. Association for Computational Linguistics.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390, Atlanta, Georgia. Association for Computational Linguistics.

Braja Gopal Patra, Dipankar Das, Amitava Das, and Rajendra Prasath. 2015. Shared task on sentiment analysis in indian languages (sail) tweets-an overview. In *International Conference on Mining Intelligence and Knowledge Exploration*, pages 650–655. Springer.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2020. Embeddings in natural language processing: Theory and advances in vector representations of meaning. *Synthesis Lectures on Human Language Technologies*, 13(4):1–175.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, and Rada Mihalcea. 2020. Beneath the tip of the iceberg: Current challenges and new directions in sentiment analysis research. *IEEE Transactions on Affective Computing*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada. Association for Computational Linguistics.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

Asahi Ushio and Jose Camacho-Collados. 2021. T-NER: An all-round python library for transformer-based named entity recognition. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 53–62, Online. Association for Computational Linguistics.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. SemEval-2018 task 3: Irony detection in English tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana. Association for Computational Linguistics.

Katrin Weller, Axel Bruns, Jean Burgess, Merja Mahrt, and Cornelius Puschmann. 2013. *Twitter and society*. Peter Lang New York.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Qi Yang, Weinan Wang, Lucas Pierce, Rajan Vaish, Xiaolin Shi, and Neil Shah. 2021. Online communication shifts in the midst of the covid-19 pandemic: A case study on snapchat. *Proceedings of the International AAAI Conference on Web and Social Media*, 15(1):830–840.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks*, pages 45–50, Valetta, Malta.

A Languages supported

In addition to English, the sentiment analysis demo (including hashtag analysis) is also available for the following languages: Amharic, Arabic, Armenian, Basque, Bengali, Bulgarian, Burmese, Catalan, Chinese, Czech, Danish, Dhivehi, Dutch, Estonian, Finnish, French, Georgian, German, Greek, Haitian, Hebrew, Hindi, Hungarian, Icelandic, Indonesian, Italian, Japanese, Kannada, Khmer, Korean, Kurdish, Lao, Latvian, Lithuanian, Malayalam, Marathi, Nepali, Norwegian, Oriya, Panjabi, Persian, Polish, Pushto, Romanian, Russian, Serbian, Sindhi, Sinhala, Slovenian, Spanish, Swedish, Tagalog, Tamil, Telegu, Thai, Turkish, Uighur, Ukranian, Urdu, Vietnamese, Welsh. These languages are supported both by a XLM-T multilingual model and the Twitter API.

B Model Links

Table 4 lists all TweetNLP models and their corresponding Hugging Face model hub links.

We release the word embeddings along with Gensim-optimized versions: (1) English-monolingual word embeddings are available at `https://tweetnlp.org/downloads/twitter-2021-124m-300d.new.bin`; (2) Multilingual word embeddings are available at `https://tweetnlp.org/downloads/twitter-multilingual-300d.new.bin`.

Table 5 lists the baselines used for the evaluation (Section 7) and their corresponding Hugging Face hub links.

Model	Link
TweetEval	https://huggingface.co/cardiffnlp/twitter-roberta-base
TimeLMs-21 (default)	https://huggingface.co/cardiffnlp/twitter-roberta-base-2021-124m
XLM-T	https://huggingface.co/cardiffnlp/twitter-xlm-roberta-base
Sentiment Analysis	https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest
Multilingual Sentiment Analysis	https://huggingface.co/cardiffnlp/twitter-xlm-roberta-base-sentiment
Emotion Recognition	https://huggingface.co/cardiffnlp/twitter-roberta-base-emotion
Emoji Prediction	https://huggingface.co/cardiffnlp/twitter-roberta-base-emoji
Irony Detection	https://huggingface.co/cardiffnlp/twitter-roberta-base-irony
Hate Speech Detection	https://huggingface.co/cardiffnlp/twitter-roberta-base-hate
Offensive Language Identification	https://huggingface.co/cardiffnlp/twitter-roberta-base-offensive
Stance Detection (abortion)	https://huggingface.co/cardiffnlp/twitter-roberta-base-stance-abortion
Topic Classification	https://huggingface.co/cardiffnlp/tweet-topic-21-multi
Named Entity Recognition	https://huggingface.co/tner/twitter-roberta-base-dec2021-tweetner7-all
Tweet Embeddings	https://huggingface.co/cambridgeltl/tweet-roberta-base-embeddings-v1

Table 4: Hugging Face model links of all the NLP models included in TweetNLP (if available).

Model	Link
RoBERTa-base	https://huggingface.co/roberta-base
XLM-R	https://huggingface.co/xlm-roberta-base
BERTweet	https://huggingface.co/vinai/bertweet-base
Sentence-BERT	https://huggingface.co/sentence-transformers/bert-base-nli-mean-tokens
all-mpnet-base-v2	https://huggingface.co/sentence-transformers/all-mpnet-base-v2
Mirror-RoBERTa	https://huggingface.co/cambridgeltl/mirror-roberta-base-sentence-drophead
SimCSE-RoBERTa	https://huggingface.co/princeton-nlp/unsup-simcse-roberta-base

Table 5: Baseline models' Hugging Face links (if available).