Analyzing Political Parody in Social Media

Antonis Maronikolakis[1*†] **Danae Sánchez Villegas**[2*]
Daniel Preoţiuc-Pietro[3] **Nikolaos Aletras**[2]
[1] Center for Information and Language Processing, LMU Munich, Germany
[2] Computer Science Department, University of Sheffield, UK
[3] Bloomberg LP
antmarakis@cis.lmu.de, {dsanchezvillegas1, n.aletras}@sheffield.ac.uk
dpreotiucpie@bloomberg.net

Abstract

Parody is a figurative device used to imitate an entity for comedic or critical purposes and represents a widespread phenomenon in social media through many popular parody accounts. In this paper, we present the first computational study of parody. We introduce a new publicly available data set of tweets from real politicians and their corresponding parody accounts. We run a battery of supervised machine learning models for automatically detecting parody tweets with an emphasis on robustness by testing on tweets from accounts unseen in training, across different genders and across countries. Our results show that political parody tweets can be predicted with an accuracy up to 90%. Finally, we identify the markers of parody through a linguistic analysis. Beyond research in linguistics and political communication, accurately and automatically detecting parody is important to improving fact checking for journalists and analytics such as sentiment analysis through filtering out parodical utterances.[1]

1 Introduction

Parody is a figurative device which is used to imitate and ridicule a particular target (Rose, 1993) and has been studied in linguistics as a figurative trope distinct to irony and satire (Kreuz and Roberts, 1993; Rossen-Knill and Henry, 1997). Traditional forms of parody include editorial cartoons, sketches or articles pretending to have been authored by the parodied person.[2] A new form of parody recently emerged in social media, and Twitter in particular, through accounts that impersonate public figures. Highfield (2016) defines parody accounts acting as: *a known, real person, for obviously comedic purposes. There should be no risk of mistaking their tweets for their subject's actual views; these accounts play with stereotypes of these figures or juxtapose their public image with a very different, behind-closed-doors persona.*

A very popular type of parody is political parody which plays an important role in public speech by offering irreverent interpretations of political personas (Hariman, 2008). Table 1 shows examples of very popular (over 50k followers) and active (thousands of tweets sent) political parody accounts on Twitter. Sample tweets show how the style and topic of parody tweets are similar to those from the real accounts, which may pose issues to automatic classification.

While closely related figurative devices such as irony and sarcasm have been extensively studied in computational linguistics (Wallace, 2015; Joshi et al., 2017), parody yet to be explored using computational methods. In this paper, we aim to bridge this gap and conduct, for the first time, a systematic study of political parody as a figurative device in social media. To this end, we make the following contributions:

1. A novel classification task where we seek to automatically classify real and parody tweets. For this task, we create a new large-scale publicly available data set containing a total of 131,666 English tweets from 184 parody accounts and corresponding real accounts of politicians from the US, UK and other countries (Section 3);
2. Experiments with feature- and neural-based machine learning models for parody detection, which achieve high predictive accuracy of up to 89.7% F1. These are focused on the robust-

[*] Equal contribution.

[†] Work was done while at the University of Sheffield.

[1] Data is available here: https://archive.org/details/parody_data_acl20

[2] The 'Kapou Opa' column by K. Maniatis that parodies Greek popular persons was a source of inspiration for this work - https://www.oneman.gr/originals/to-imerologio-karantinas-tou-dimitri-koutsoumpa/

ness of classification, with test data from: a) users; b) genders; c) locations; unseen in training (Section 5);

3. Linguistic analysis of the markers of parody tweets and of the model errors (Section 6).

We argue that understanding the expression and use of parody in natural language and automatically identifying it are important to applications in computational social science and beyond. Parody tweets can often be misinterpreted as facts even though Twitter only allows parody accounts if they are explicitly marked as parody[3] and the poster does not have the intention to mislead. For example, the Speaker of the US House of Representatives, Nancy Pelosi, falsely cited a Michael Flynn parody tweet;[4] and many users were fooled by a Donald Trump parody tweet about 'Dow Joans'.[5] Thus, accurate parody classification methods can be useful in downstream NLP applications such as automatic fact checking (Vlachos and Riedel, 2014) and rumour verification (Karmakharm et al., 2019), sentiment analysis (Pang et al., 2008) or nowcasting voting intention (Tumasjan et al., 2010; Lampos et al., 2013; Tsakalidis et al., 2018).

Beyond NLP, parody detection can be used in: (i) political communication, to study and understand the effects of political parody in the public speech on a large scale (Hariman, 2008; Highfield, 2016); (ii) linguistics, to identify characteristics of figurative language (Rose, 1993; Kreuz and Roberts, 1993; Rossen-Knill and Henry, 1997); (iii) network science, to identify the adoption and diffusion mechanisms of parody (Vosoughi et al., 2018).

2 Related Work

Parody in Linguistics Parody is an artistic form and literary genre that dates back to Aristophanes in ancient Greece who parodied argumentation styles in *Frogs*. Verbal parody was studied in linguistics as a figurative trope distinct to irony and satire (Kreuz and Roberts, 1993; Rossen-Knill and Henry, 1997) and researchers long debated its definition and theoretic distinctions to other types of humor (Grice et al., 1975; Sperber, 1984; Wilson, 2006; Dynel, 2014). In general, verbal parody involves a highly situated, intentional, and conventional speech act (Rossen-Knill and Henry, 1997) composed of both a negative evaluation and a form of pretense or echoic mention (Sperber, 1984; Wilson, 2006; Dynel, 2014) through which an entity is mimicked or imitated with the goal of criticizing it to a comedic effect. Thus, imitative composition for amusing purpose is an an inherent characteristic of parody (Franke, 1971). The parodist intentionally re-presents the object of the parody and flaunts this re-presentation (Rossen-Knill and Henry, 1997).

Parody on Social Media Parody is considered an integral part of Twitter (Vis, 2013) and previous studies on parody in social media focused on analysing how these accounts contribute to topical discussions (Highfield, 2016) and the relationship between identity, impersonation and authenticity (Page, 2014). Public relation studies showed that parody accounts impact organisations during crises while they can become a threat to their reputation (Wan et al., 2015).

Satire Most related to parody, satire has been tangentially studied as one of several prediction targets in NLP in the context of identifying disinformation (McHardy et al., 2019; de Morais et al., 2019). In (Rashkin et al., 2017), the authors evaluate the language of real news with that of satire, hoaxes, and propaganda in order to identify linguistic features of unreliable text. They demonstrate how stylistic characteristics can help to decide the text's veracity. The study of parody is therefore relevant to this topic, as satire and parodies are classified by some as a type of disinformation with 'no intention to cause harm but has potential to fool' (Wardle and Derakhshan, 2018).

Irony and Sarcasm There is a rich body of work in NLP on identifying irony and sarcasm as a classification task (Wallace, 2015; Joshi et al., 2017). Van Hee et al. (2018) organized two open shared tasks. The first aims to automatically classify tweets as ironic or not, and the second is on identifying the type of irony expressed in tweets. However, the definition of irony is usually 'a trope whose actual meaning differs from what is literally enunciated' (Van Hee et al., 2018), following the Gricean belief that the hallmark of irony is to communicate the opposite of the literal meaning (Wilson, 2006), violating the first maxim of Quality (Grice et al., 1975). In this

[3] Both the profile description and account name need to mention this – https://help.twitter.com/en/rules-and-policies/parody-account-policy

[4] https://tinyurl.com/ybbrh74g

[5] https://tinyurl.com/s34dwgm

Account type	Twitter Handle	Sample tweet
Real	`@realDonaldTrump`	The Republican Party, and me, had a GREAT day yesterday with respect to the phony Impeachment Hoax, & yet, when I got home to the White House & checked out the news coverage on much of television, you would have no idea they were reporting on the same event. FAKE & CORRUPT NEWS!
Parody	`@realDonaldTrFan`	Lies! Kampala Harris says my crimes are committed in plane site! Shes lying! My crimes are ALWAYS hidden! ALWAYS!!
Real	`@BorisJohnson`	Our NHS will never be on the table for any trade negotiations. Were investing more than ever before - and when we leave the EU, we will introduce an Australian style, points-based immigration system so the NHS can plan for the future.
Parody	`@BorisJohnson_MP`	People seem to be ignoring the many advantages of selling off the NHS, like the fact that hospitals will be far more spacious once poor people can't afford to use them.

Table 1: Two examples of Twitter accounts of politicians and their corresponding parody account with a sample tweet from each.

sense, irony is treated in NLP in a similar way as sarcasm (González-Ibáñez et al., 2011; Khattri et al., 2015; Joshi et al., 2017). In addition to the words in the utterance, further using the user and pragmatic context is known to be informative for irony or sarcasm detection in NLP (Bamman and Smith, 2015; Wallace, 2015). For instance, Oprea and Magdy (2019) make use of user embeddings for textual sarcasm detection. In the design of our data splits, we aim to limit the contribution of this aspects from the results.

Relation to other NLP Tasks The pretense aspect of parody relates our task to a few other NLP tasks. In authorship attribution, the goal is to predict the author of a given text (Stamatatos, 2009; Juola et al., 2008; Koppel et al., 2009). However, there is no intent for the authors to imitate the style of others and most differences between authors are in the topics they write about, which we aim to limit by focusing on political parody. Further, in our setups, no tweets from an author are in both training and testing to limit the impact of terms specific to a particular person.

Pastiche detection (Dinu et al., 2012) aims to distinguish between an original text and a text written by someone aiming to imitate the style of the original author with the goal of impersonating. Most similar in experimental setup to our task, Preoţiuc-Pietro and Devlin Marier (2019) aim to distinguish between tweets published from the same account by different types of users: politicians or their staff. While both pastiches and staff writers aim to present similar content with similar style to the original authors, the texts lack the humorous component specific of parodies.

A large body of related NLP work has explored the inference of user characteristics. Past research studied predicting the type of a Twitter account, most frequently between individual or organizational, using linguistic features (De Choudhury et al., 2012; McCorriston et al., 2015; Mac Kim et al., 2017). A broad literature has been devoted to predicting personal traits from language use on Twitter, such as gender (Burger et al., 2011), age (Nguyen et al., 2011), geolocation (Cheng et al., 2010), political preference (Volkova et al., 2014; Preoţiuc-Pietro et al., 2017), income (Preoţiuc-Pietro et al., 2015; Aletras and Chamberlain, 2018), impact (Lampos et al., 2014), socio-economic status (Lampos et al., 2016), race (Preoţiuc-Pietro and Ungar, 2018) or personality (Schwartz et al., 2013; Preoţiuc-Pietro et al., 2016).

3 Task & Data

We define parody detection in social media as a binary classification task performed at the social media post level. Given a post T, defined as a sequence of tokens $T = \{t_1, ..., t_n\}$, the aim is to label T either as parody or genuine. Note that one could use social network information but this is out of the paper's scope as we only focus on parody as a linguistic device.

We create a new publicly available data set to study this task, as no other data set is available. We perform our analysis on a set of users from the same domain (politics) to limit variations caused by topic. We first identify real and parody accounts of politicians on Twitter posting in English from the United States of America (US), the United Kingdom (UK) and other accounts posting in English from the rest of the world. We opted to use

Twitter because it is arguably the most popular platform for politicians to interact with the public or with other politicians (Parmelee and Bichard, 2011). For example, 67% of prospective parliamentary candidates for the 2019 UK general election have an active Twitter account.[6] Twitter also allows to maintain parody accounts, subject to adding explicit markers in both the user bio and handle such as `parody`, `fake`.[7] Finally, we label tweets as parody or real, depending on the type of account they were posted from. We highlight that we are not using user description or handle names in prediction, as this would make the task trivial.

3.1 Collecting Real and Parody Politician Accounts

We first query the public Twitter API using the following terms: {`parody`, `#parody`, `parody account`, `fake`, `#fake`, `fake account`, `not real`} to retrieve candidate parody accounts according to Twitter's policy. From that set, we exclude any accounts also matching `fan` or `commentary` in their bio or account name since these are likely to be not posting parodical content. We also exclude private and deactivated accounts and accounts with a majority of non-English tweets.

After collecting this initial set of parody candidates, the authors of the paper manually inspected up to the first ten original tweets from each candidate to identify whether an account is a parody or not following the definition of a public figure parody account from Highfield (2016) (see Section 1), further filtering out non-parody accounts. We keep a single parody account in case of multiple parody accounts about the same person. Finally, for each remaining account, the authors manually identified the corresponding real politician account in order to collect pairs of real and parody, if possible.

Following the process above, we were able to identify parody accounts of 103 unique people, with 81 having a corresponding real account. The authors also identified the binary gender and location (country) of the accounts using publicly available records. This resulted in 21.6% female accounts (women parliamentarians percentages as of 2017: 19% US, 30% UK, 28.8% OECD average).[8] The majority of the politicians are located in the US (44.5%) followed by the UK (26.7%) while 28.8% are from the rest of the world (e.g. Germany, Canada, India, Russia).

3.2 Collecting Real and Parody Tweets

We collect all of the available original tweets, excluding retweets and quoted tweets, from all the parody and real politician accounts.[9] We further balance the number of tweets in a real – parody account pair in order for our experiments and linguistic analysis not to be driven by a few prolific users or by imbalances in the tweet ratio for a specific pair. We keep a ratio of maximum ±20% between the real and parody tweets per pair by keeping all tweets from the less prolific account and randomly down-sampling from the more prolific one. Subsequently, for the parody accounts with no corresponding real account, we sample a number of tweets equal to the median number of tweets for the real accounts. Finally, we label tweets as parody or real, depending on the type of account they come from. In total, the data set contains 131,666 tweets, with 65,710 real and 65,956 parody.

3.3 Data Splits

To test that automatically predicting political parody is robust and generalizes to held-out situations not included in the training data, we create the following three data splits for running experiments:

Person Split We first split the data by adding all tweets from each real – parody account pair to a single split, either train, development or test. To obtain a fairly balanced data set without pairs of accounts with a large number of tweets dominating any splits, we compute the mean between real and parody tweets for each account, and stratify them, with pairs of proportionally distributed means across the train, development, and test sets (see Table 2).

Gender Split We also split the data by the gender of the politicians into training, development and test, obtaining two versions of the data: (i) one with female accounts in train/dev and male in test; and (ii) male accounts in train/dev and female in test (see Table 3).

[6] https://www.mpsontwitter.co.uk/

[7] https://help.twitter.com/en/rules-and-policies/parody-account-policy

[8] https://data.oecd.org/inequality/women-in-politics.htm

[9] Up to maximum 3200 tweets/account according to Twitter API restrictions.

Person					
	Train	**Dev**	**Test**	**Total**	**Avg. tokens (Train)**
Real	51,460	6,164	8,086	65,710	23.33
Parody	51,706	6,164	8,086	65,956	20.15
All	103,166	12,328	16,172	131,666	22.55

Table 2: Data set statistics with the person split.

Gender				
Trained on		**Real**	**Parody**	**Total**
Female	**Train**	10,081	11,036	21,117
	Dev	302	230	532
	Test (Male)	55,327	54,690	110,017
Male	**Train**	51,048	50,184	101,232
	Dev	4,279	4,506	8,785
	Test (Female)	10,383	11,266	21,649

Table 3: Data set statistics with the gender split (Male, Female).

Location				
Trained on		**Real**	**Parody**	**Total**
US & RoW	**Train**	47,018	45,005	92,023
	Dev	1,030	2,190	3,220
	Test (UK)	17,662	18,761	36,423
UK & RoW	**Train**	33,687	35,371	69,058
	Dev	1,030	1,274	2,304
	Test (US)	30,993	29,311	60,304
US & UK	**Train**	43,211	42,597	85,808
	Dev	5,444	5,475	10,919
	Test (RoW)	17,055	17,884	34,939

Table 4: Data set statistics with the location split (US, UK, Rest of the World–RoW).

Location split Finally, we split the data based on the location of the politicians. We group the accounts in three groups of locations: US, UK and the rest of the world (**RoW**). We obtain three different splits, where each group makes up the test set and the other two groups make up the train and development set (see Table 4).

3.4 Text Preprocessing

We preprocess text by lower-casing, replacing all URLs and anonymizing all mentions of usernames with placeholder token. We preserve emoticons and punctuation marks and replace tokens that appear in less than five tweets with a special 'unknown' token. We tokenize text using DLATK (Schwartz et al., 2017), a Twitter-aware tokenizer.

4 Predictive Models

We experiment with a series of approaches to classification of parody tweets, ranging from linear models, neural network architectures and pre-trained contextual embedding models. Hyperparameter selection is included in the Appendix.

4.1 Linear Baselines

LR-BOW As a first baseline, we use a logistic regression with standard bag-of-words (LR-BOW) representation of the tweets.

LR-BOW+POS We extend LR-BOW using syntactic information from Part-Of-Speech (POS) tags. We first tag all tweets in our data using the NLTK tagger and then we extract bag-of-words features where each unigram consists of a token with its associated POS tag.

4.2 BiLSTM-Att

The first neural model is a bidirectional Long-Short Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) with a self-attention mechanism (BiLSTM-Att; Zhou et al. (2016)). Tokens t_i in a given tweet $T = \{t_1, ..., t_n\}$ are mapped to embeddings and passed through a bidirectional LSTM. A single tweet representation (h) is computed as the sum of the resulting contextualized vector representations $(\sum_i a_i h_i)$ where a_i is the self-attention score in timestep i. The tweet representation (h) is subsequently passed to the output layer using a sigmoid activation function.

4.3 ULMFit

The Universal Language Model Fine-tuning (ULMFit) is a method for efficient transfer learning (Howard and Ruder, 2018). The key intuition is to train a text encoder on a language modelling task (i.e. predicting the next token in a sequence) where data is abundant, then fine-tune it on a target task where data is more limited. During fine-tuning, ULMFit uses gradual layer unfreezing to avoid catastrophic forgetting. We experiment with using AWD-LSTM (Merity et al., 2018) as the base text encoder pretrained on the Wikitext 103 data set and we fine-tune it on our own parody classification task. For this purpose, after the AWS-LSTM layers, we add a fully-connected layer with a ReLU activation function followed by an output layer with a sigmoid activation function. Before each of these two additional layers, we perform batch normalization.

4.4 BERT and RoBERTa

Bidirectional Encoder Representations from Transformers (BERT) is a language model based on transformer networks (Vaswani et al., 2017) pre-trained on large corpora (Devlin et al., 2019). The model makes use of multiple multi-head attention layers to learn bidirectional embeddings for input tokens. It is trained for masked language modelling, where a fraction of the input tokens in a given sequence are masked and the task is to predict a masked word given its context. BERT uses wordpieces which are passed through an embedding layer and get summed together with positional and segment embeddings. The former introduce positional information to the attention layers, while the latter contain information about the location of a segment. Similar to ULMFit, we fine-tune the BERT-base model for predicting parody tweets by adding an output dense layer for binary classification and feeding it with the 'classification' token.

We further experiment with RoBERTa (Liu et al., 2019), which is an extenstion of BERT trained on more data and different hyperparameters. RoBERTa has been showed to improve performance in various benchmarks compared to the original BERT (Liu et al., 2019).

4.5 XLNet

XLNet is another pre-trained neural language model based on transformer networks (Yang et al., 2019). XLNet is similar to BERT in its structure, but is trained on a permutated (instead of masked) language modelling task. During training, sentence words are permuted and the model predicts a word given the shuffled context. We also adapt XLNet for predicting parody, similar to BERT and ULMFit.

4.6 Model Hyperparameters

We optimize all model parameters on the development set for each data split (see Section 3).

Linear models For the LR-BOW, we use n-grams with n = (1, 2), $n \in$ {(1, 1), (1, 2), (1, 3) weighted by TF.IDF. For the LR-BOW+POS, we use TF with POS n-grams where n = (1, 3). For both baselines we use L2 regularization.

BiLSTM-Att We use 200-dimensional GloVe embeddings (Pennington et al., 2014) pre-trained on Twitter data. The maximum sequence length is set to 50 covering 95% of the tweets in the training set. The LSTM size is $h = 300$ where $h \in \{50, 100, 300\}$ with dropout $d = 0.5$ where $d \in \{.2, .5\}$. We use Adam (Kingma and Ba, 2014) with default learning rate, minimizing the binary cross-entropy using a batch size of 64 over 10 epochs with early stopping.

ULMFit We first update only the AWD-LSTM weights with a learning rate l = 2e-3 for one epoch where $l \in$ {1e-3, 2e-3, 4e-3} for language modeling. Then, we update both the AWD-LSTM and embedding weights for one more epoch, using a learning rate of l = 2e-5 where $l \in$ {1e-4, 2e-5, 5e-5}. The size of the intermediate fully-connected layer (after AWD-LSTM and before the output) is set by default to 50. Both in the intermediate and output layers we use default dropout of 0.08 and 0.1 respectively from Howard and Ruder (2018).

BERT and RoBERTa For BERT, we used the base model (12 layers and 110M total parameters) trained on lowercase English. We fine-tune it for 1 epoch with a learning rate l = 5e-5 where $l \in$ {2e-5, 3e-5, 5e-5} as recommended in Devlin et al. (2019) with a batch size of 128. For RoBERTa, we use the same fine-tuning parameters as BERT.

XLNet We use the same parameters as BERT except for the learning rate, which we set at l = 4e-5 where $l \in$ {2e-5, 4e-5, 5e-5}.

5 Results

This section contains the experimental results obtained on all three different data splits proposed in Section 3. We evaluate our methods (Section 4) using several metrics, including accuracy, precision, recall, macro F1 score, and Area under the ROC (AUC). We report results over three runs using different random seeds and we report the average and standard deviation.

5.1 Person Split

Table 5 presents the results for the parody prediction models with the data split by person. We observe the architectures using pre-trained text encoders (i.e. ULMFit, BERT, RoBERTa and XLNet) outperform both neural (BiLSTM-Att) and feature-based (LR-BOW and LR-BOW+POS) by a large margin across metrics with transformer architectures (BERT, RoBERTa and XLNet) performing best. The highest scoring model,

Person					
Model	**Acc**	**P**	**R**	**F1**	**AUC**
LR-BOW	73.95 ±0.00	70.08 ± 0.01	83.53 ±0.02	76.19 ±0.00	73.96 ±0.00
LR-BOW+POS	74.33 ±0.00	71.34 ±0.00	81.19 ±0.00	75.95 ±0.00	74.34 ±0.00
BiLSTM-Att	79.92 ±0.01	81.63 ±0.01	77.11 ±0.03	79.29 ±0.02	79.91 ±0.01
ULMFit	81.11 ±0.38	75.57 ±2.03	84.97 ±0.87	81.05 ±0.42	81.10 ±0.38
BERT	87.65 ±0.29	87.63 ±0.58	87.67 ±0.40	87.65 ±0.18	87.65 ±0.32
RoBERTa	**90.01 ±0.35**	**90.90 ±0.55**	**88.45 ±0.22**	**89.66 ±0.33**	**90.05 ±0.29**
XLNet	86.45 ±0.41	88.24 ±0.52	85.18 ±0.40	86.68 ±0.37	86.45 ±0.36

Table 5: Accuracy (Acc), Precision (P), Recall (R), F1-Score (F1) and ROC-AUC for parody prediction splitting by person (± std. dev.). Best results are in bold.

RoBERTa, classifies accounts (parody and real) with an accuracy of 90, which is more than 8% greater than the best non-transformer model (the ULMFit method). RoBERTa also outperforms the Logistic Regression baselines (LR-BOW and LR-BOW+POS) by more than 16 in accuracy and 13 in F1 score. Furthermore, it is the only model to score higher than 90 on precision.

Gender		
Model	**M→F**	**F→M**
LR-BOW	78.89	76.63
LR-BOW+POS	78.74	76.74
BiLSTM-Att	77.00	77.11
ULMFit	81.20	82.53
BERT	85.85	84.40
RoBERTa	**87.11**	**84.87**
XLNet	85.69	84.16

Table 6: F1-scores for parody prediction splitting by gender (Male-M, Female-F). Best results are in bold.

Location			
Model	🇺🇸 + 🇬🇧 → 🌍	🌍 + 🇺🇸 → 🇬🇧	🌍 + 🇬🇧 → 🇺🇸
LR-BOW	78.58	78.27	77.97
LR-BOW+POS	78.27	77.88	78.08
BiLSTM-Att	80.29	77.59	73.19
ULMFit	83.47	81.55	81.55
BERT	86.69	83.78	83.12
RoBERTa	**87.70**	**85.10**	**85.99**
XLNet	85.32	**85.17**	85.32

Table 7: F1-scores for parody prediction splitting by location. Best results are in bold.

5.2 Gender Split

Table 6 shows the F1-scores obtained when training on the gender splits, i.e. training on male and testing on female accounts and vice versa. We first observe that models trained on the male set are in general more accurate than models trained on the female set, with the sole exception of ULMFit. This is probably due to the fact that the data set is imbalanced towards men as shown in Table 3 (see also Section 3). We also do not observe a dramatic performance drop compared to the mixed-gender model on the person split (see Table 5). Again, RoBERTa is the most accurate model when trained in both splits, obtaining an F1-score of 87.11 and 84.87 for the male and female data respectively. The transformer-based architectures are again the best performing models overall, but the difference between them and the feature-based methods is smaller than it was on the person split.

5.3 Location Split

Table 7 shows the F1-scores obtained training our models on the location splits: (i) train/dev on UK and RoW, test on US; (ii) train/dev on US and RoW, test on UK; and (iii) train/dev on US and UK, test on RoW. In general, the best results are obtained by training on the US & UK split, while results of the models trained on the RoW & US, and RoW & UK splits are similar. The model with the best performance trained on US & UK, and RoW & UK splits is RoBERTa with F1 scores of 87.70 and 85.99 respectively. XLNet performs slightly better than RoBERTa when trained on RoW & US data split.

5.4 Discussion

Through experiments over three different data splits, we show that all models predict parody tweets consistently above random, even if tested on people unseen in training. In general, we observe that the pre-trained contextual embedding based models perform best, with an average of around 10 F1 better than the linear methods. From these methods, we find that RoBERTa outperforms the other methods by a small, but consistent margin, similar to past research (Liu et al., 2019). Further, we see that the predictions are robust to any location or gender specific differences, as the performance on held-out locations and genders are close to when splitting by person with a maximum of < 5 F1 drop, also impacted by training on less data (e.g. female users). This highlights the fact that our models capture information beyond topics or features specific to any person, gender or location and can potentially identify stylistic differences between parody and real tweets.

6 Analysis

We finally perform an analysis based on our novel data set to uncover the peculiarities of political parody and understand the limits of the predictive models.

6.1 Linguistic Feature Analysis

We first analyse the linguistic features specific of real and parody tweets. For this purpose, we use the method introduced in (Schwartz et al., 2013) and used in several other analyses of user traits (Preoţiuc-Pietro et al., 2017) or speech acts (Preoţiuc-Pietro et al., 2019). We thus rank the feature sets described in Section 4 using univariate Pearson correlation (note that for the analysis we use POS tags instead of POS n-grams). Features are normalized to sum up to unit for each tweet. Then, for each feature, we compute correlations independently between its distribution across posts and the label of the post (parody or not).

Real		Parody	
Feature	r	Feature	r
Unigrams			
our	0.140	i	0.181
in	0.131	?	0.156
and	0.129	<mention>	0.145
:	0.118	me	0.136
&	0.114	not	0.106
today	0.105	like	0.097
to	0.105	my	0.095
of	0.098	dude	0.094
the	0.091	don’t	0.090
at	0.087	i’m	0.087
lhl	0.086	just	0.083
great	0.085	know	0.081
with	0.084	#feeltheburp	0.078
de	0.079	you	0.076
meeting	0.078	#callmedick	0.075
for	0.077	#imwithme	0.073
across	0.073	”	0.073
families	0.073	#visionzero	0.069
on	0.070	if	0.069
country	0.067	have	0.067
POS (Unigrams and Bigrams)			
NN IN	0.1600	RB	0.1749
IN	0.1507	PRP	0.1546
CC	0.1309	RB VB	0.1271
IN JJ	0.1210	VBP	0.1206
NNS IN	0.1165	VBP RB	0.1123
NN CC	0.1114	.	0.1114
IN NN	0.1048	NNP NNP	0.1094
NN TO	0.1030	NN NNP	0.1057
NNS TO	0.1013	WRB	0.0925
TO	0.1001	VBP PRP	0.0904
CC JJ	0.0972	IN PRP	0.0890
IN DT	0.0941	NN VBP	0.0863
: JJ	0.0875	RB .	0.0854
NNS	0.0855	NNP	0.0837
: NN	0.0827	JJ VBP	0.0813

Table 8: Feature correlations with parody and real tweets, sorted by Pearson correlation (r). All correlations are significant at $p < .01$, two-tailed t-test.

Table 8 presents the top unigrams and part-of-speech features correlated with real and parody tweets. We first note that the top features related to either parody or genuine tweets are function words or related to style, as opposed to the topic. This enforces that the make-up of the data set or any of its categories are not impacted by topic choice and parody detection is mostly a stylistic difference. The only exception are a few hashtags related to parody accounts (e.g. #imwithme), but on a closer inspection, all of these are related to tweets from a single parody account and are thus not useful in prediction by any setup, as tweets containing these will only appear in either the train or test set.

The top features related to either category of tweets are pronouns (‘our’ for genuine tweets, ‘i’ for parody tweets). In general, we observe that parody tweets are much more personal and include possessives (‘me’, ‘my’, ‘i’, “i’m”, PRP) or second person pronouns (‘you’). This indicates that parodies are more personal and direct, which is also supported by use of more @-mentions and quotation marks. The real politician tweets are more impersonal and the use of 'our' indicates a desire to include the reader in the conversation.

The real politician tweets include more stopwords (e.g. prepositions, conjunctions, determiners), which indicate that these tweets are more well formed. Conversely, the parody tweets include more contractions ("don't", "i'm"), hinting to a less formal style ('dude'). Politician tweets frequently use their account to promote events they participate in or are relevant to the day-to-day schedule of a politician, as hinted by several prepositions ('at', 'on') and words ('meeting', "today') (Preoţiuc-Pietro and Devlin Marier, 2019). For example, this is a tweet of the U.S. Senator from Connecticut, Chris Murphy:

> *Rudy Giuliani is in Ukraine* ***today, meeting*** *with Ukranian leaders on behalf of the President of the United States, representing the President's re-election campaign.[...]*

Through part-of-speech patterns, we observe that parody accounts are more likely to use verbs in the present singular (VBZ, VBP). This hints that parody tweets explicitly try to mimic direct quotes from the parodied politician in first person and using present tense verbs, while actual politician tweets are more impersonal. Adverbs (RB) are used predominantly in parodies and a common sequence in parody tweets is adverbs followed by verbs (RB VB) which can be used to emphasize actions or relevant events. For example, the following is a tweet of a parody account (@Queen_Europe) of Angela Merkel:

> *I mean, the Brexit Express* ***literally appears*** *to be going backwards but OK <url>*

6.2 Error Analysis

Finally, we perform an error analysis to examine the behavior of our best performing model (RoBERTa) and identify potential limitations of the current approaches. The first example is a tweet by the former US president Barack Obama which was classified as parody while it is in fact a real tweet:

> *Summer's almost over, Senate Leaders. #doyourjob <url>*

Similarly, the next tweet was posted by the real account of the Virginia governor, Ralph Northam:

> *At this point, the list of Virginians Ed Gillespie *hasn't* sold out is shorter than the folks he has. <url>*

Both of the tweets above contain humoristic elements and come off as confrontational, aimed at someone else which is more prevalent in parody. We hypothesize that the model picked up this information to classify these tweets as parody. From the previous analyses, we noticed that tweets by real politicians often convey information in a more neutral or impersonal way. On the other hand, the following tweet was posted by a Mitt Romney parody account and was classified as real:

> *It's up to you, America: do you want a repeat of the last four years, or four years staggeringly worse than the last four years?*

This parody tweet, even though it is more opinionated, is more similar in style to a slogan or campaign speech and is therefore missclassified. Lastly, the following is a tweet from former President Obama that was misclassified as parody:

> *It's the #GimmeFive challenge, presidential style. <url>*

The reason behind is that there are politicians, such as Barack Obama, who often write in an informal manner and this may cause the models to misclassify this kind of tweets.

7 Conclusion

We presented the first study of parody using methods from computational linguistics and machine learning, a related but distinct linguistic phenomenon to irony and sarcasm. Focusing on political parody in social media, we introduced a freely available large-scale data set containing a total of 131,666 English tweets from 184 real and corresponding parody accounts. We defined parody prediction as a new binary classification task at a tweet level and evaluated a battery of feature-based and neural models achieving high predictive accuracy of up to 89.7% F1 on tweets from people unseen in training.

In the future, we plan to study more in depth the stylistic and figurative devices used for parody, extend the data set beyond the political case study and explore human behavior regarding parody, including how this is detected and diffused through social media.

Acknowledgments

We thank Bekah Hampson for providing early input and helping with the data annotation. NA is supported by ESRC grant ES/T012714/1 and an Amazon AWS Cloud Credits for Research Award.

References

Nikolaos Aletras and Benjamin Paul Chamberlain. 2018. Predicting Twitter user socioeconomic attributes with network and language information. In *Proceedings of the 29th on Hypertext and Social Media*, pages 20–24.

David Bamman and Noah A Smith. 2015. Contextualized Sarcasm Detection on Twitter. In *Ninth International AAAI Conference on Web and Social Media*, ICWSM, pages 574–577.

John D Burger, John Henderson, George Kim, and Guido Zarrella. 2011. Discriminating gender on Twitter. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1301–1309. Association for Computational Linguistics.

Zhiyuan Cheng, James Caverlee, and Kyumin Lee. 2010. You are where you tweet: a content-based approach to geo-locating Twitter users. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 759–768.

Munmun De Choudhury, Nicholas Diakopoulos, and Mor Naaman. 2012. Unfolding the event landscape on Twitter: Classification and exploration of user categories. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW, pages 241–244.

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.

Liviu P. Dinu, Vlad Niculae, and Maria-Octavia Sulea. 2012. Pastiche detection based on stopword rankings. Exposing Impersonators of a Romanian writer. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 72–77, Avignon, France. Association for Computational Linguistics.

Marta Dynel. 2014. Isn't it ironic? Defining the scope of humorous irony. *Humor*, 27(4):619–639.

Herbert Franke. 1971. A note on parody in Chinese traditional literature. *Oriens Extremus*, 18(2):237–251.

Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in Twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586.

H Paul Grice, Peter Cole, Jerry Morgan, et al. 1975. Logic and conversation. *1975*, pages 41–58.

Robert Hariman. 2008. Political Parody and Public Culture. *Quarterly Journal of Speech*, 94(3):247–272.

Tim Highfield. 2016. News via Voldemort: Parody accounts in topical discussions on Twitter. *New Media & Society*, 18(9):2028–2045.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339.

Aditya Joshi, Pushpak Bhattacharyya, and Mark J Carman. 2017. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):73.

Patrick Juola et al. 2008. Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3):233–334.

Twin Karmakharm, Nikolaos Aletras, and Kalina Bontcheva. 2019. Journalist-in-the-loop: Continuous learning as a service for rumour analysis. 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): System Demonstrations*, pages 115–120.

Anupam Khattri, Aditya Joshi, Pushpak Bhattacharyya, and Mark Carman. 2015. Your sentiment precedes you: Using an author's historical tweets to predict sarcasm. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 25–30.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Moshe Koppel, Jonathan Schler, and Shlomo Argamon. 2009. Computational methods in authorship attribution. *Journal of the Association for Information Science and Technology*, 60(1):9–26.

Roger J. Kreuz and Richard M. Roberts. 1993. On satire and parody: The importance of being ironic. *Metaphor and Symbolic Activity*, 8(2):97–109.

Vasileios Lampos, Nikolaos Aletras, Jens K Geyti, Bin Zou, and Ingemar J Cox. 2016. Inferring the socioeconomic status of social media users based on behaviour and language. In *ECIR*, pages 689–695.

Vasileios Lampos, Nikolaos Aletras, Daniel Preoţiuc-Pietro, and Trevor Cohn. 2014. Predicting and characterising user impact on Twitter. In *14th conference of the European chapter of the Association for*

Computational Linguistics 2014, EACL 2014, pages 405–413.

Vasileios Lampos, Daniel Preoţiuc-Pietro, and Trevor Cohn. 2013. A user-centric model of voting intention from social media. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 993–1003, Sofia, Bulgaria. 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*.

Sunghwan Mac Kim, Qiongkai Xu, Lizhen Qu, Stephen Wan, and Cécile Paris. 2017. Demographic Inference on Twitter using Recursive Neural Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 471–477.

James McCorriston, David Jurgens, and Derek Ruths. 2015. Organizations are users too: Characterizing and detecting the presence of organizations on Twitter. ICWSM, pages 650–653.

Robert McHardy, Heike Adel, and Roman Klinger. 2019. Adversarial training for satire detection: Controlling for confounding variables. 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 660–665.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018. Regularizing and optimizing LSTM language models. In *International Conference on Learning Representations*.

Janaína Ignácio de Morais, Hugo Queiroz Abonizio, Gabriel Marques Tavares, André Azevedo da Fonseca, and Sylvio Barbon Jr. 2019. Deciding among fake, satirical, objective and legitimate news: A multi-label classification system. In *Proceedings of the XV Brazilian Symposium on Information Systems*, page 22.

Dong Nguyen, Noah A Smith, and Carolyn P Rosé. 2011. Author age prediction from text using linear regression. In *Proceedings of the 5th ACL-HLT workshop on language technology for cultural heritage, social sciences, and humanities*, pages 115–123. Association for Computational Linguistics.

Silviu Oprea and Walid Magdy. 2019. Exploring author context for detecting intended vs perceived sarcasm. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2859.

Ruth Page. 2014. *Hoaxes, hacking and humour: analysing impersonated identity on social network sites*, pages 46–64. Palgrave Macmillan UK.

Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135.

John H Parmelee and Shannon L Bichard. 2011. *Politics and the Twitter revolution: How tweets influence the relationship between political leaders and the public*. Lexington Books.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Daniel Preoţiuc-Pietro, Jordan Carpenter, Salvatore Giorgi, and Lyle Ungar. 2016. Studying the dark triad of personality through Twitter behavior. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 761–770.

Daniel Preoţiuc-Pietro and Rita Devlin Marier. 2019. Analyzing linguistic differences between owner and staff attributed tweets. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2848–2853.

Daniel Preoţiuc-Pietro, Mihaela Gaman, and Nikolaos Aletras. 2019. Automatically identifying complaints in social media. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5019.

Daniel Preoţiuc-Pietro, Ye Liu, Daniel Hopkins, and Lyle Ungar. 2017. Beyond binary labels: political ideology prediction of Twitter users. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 729–740.

Daniel Preoţiuc-Pietro and Lyle Ungar. 2018. User-level race and ethnicity predictors from Twitter text. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1534–1545.

Daniel Preoţiuc-Pietro, Svitlana Volkova, Vasileios Lampos, Yoram Bachrach, and Nikolaos Aletras. 2015. Studying user income through language, behaviour and affect in social media. *PloS one*, 10(9).

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937.

Margaret A Rose. 1993. *Parody: ancient, modern and post-modern*. Cambridge University Press.

Deborah F. Rossen-Knill and Richard Henry. 1997. The pragmatics of verbal parody. *Journal of Pragmatics*, 27(6):719 – 752.

H Andrew Schwartz, Johannes C Eichstaedt, Margaret L Kern, Lukasz Dziurzynski, Stephanie M Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin EP Seligman, et al. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):e73791.

H. Andrew Schwartz, Salvatore Giorgi, Maarten Sap, Patrick Crutchley, Lyle Ungar, and Johannes Eichstaedt. 2017. DLATK: Differential language analysis ToolKit. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 55–60.

Dan Sperber. 1984. *Verbal Irony: Pretense or Echoic Mention?* American Psychological Association.

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the Association for Information Science and Technology*, 60(3):538–556.

Adam Tsakalidis, Nikolaos Aletras, Alexandra I Cristea, and Maria Liakata. 2018. Nowcasting the stance of social media users in a sudden vote: The case of the Greek Referendum. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 367–376.

Andranik Tumasjan, Timm O Sprenger, Philipp G Sandner, and Isabell M Welpe. 2010. Predicting elections with Twitter: What 140 characters reveal about political sentiment. In *4th International AAAI Conference on Weblogs and Social Media*, pages 178–185.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Farida Vis. 2013. Twitter as a reporting tool for breaking news: Journalists tweeting the 2011 UK riots. *Digital journalism*, 1(1):27–47.

Andreas Vlachos and Sebastian Riedel. 2014. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 18–22.

Svitlana Volkova, Glen Coppersmith, and Benjamin Van Durme. 2014. Inferring user political preferences from streaming communications. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 186–196.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science*, 359(6380):1146–1151.

Byron C Wallace. 2015. Computational irony: A survey and new perspectives. *Artificial Intelligence Review*, 43(4):467–483.

Sarah Wan, Regina Koh, Andrew Ong, and Augustine Pang. 2015. Parody social media accounts: Influence and impact on organizations during crisis. *Public Relations Review*, 41(3):381 – 385.

Claire Wardle and Hossein Derakhshan. 2018. Thinking about information disorder: formats of misinformation, disinformation, and mal-information. *Journalism, fake news & disinformation. Paris: Unesco*, pages 43–54.

Deirdre Wilson. 2006. The pragmatics of verbal irony: Echo or pretence? *Lingua*, 116(10):1722–1743.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212.