2022 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT) | 978-1-6654-6090-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/BDCAT56447.2022.00042

Data Quality and Linguistic Cues for Domain-independent Deception Detection

Casey Hanks
Department of CSEE
University of Maryland, Baltimore County
Baltimore, United States
chanks1@umbc.edu

Rakesh M. Verma
Department of Computer Science
University of Houston
Houston, United States
rmverma2@central.uh.edu

***Abstract*—Deception is pervasive in today's connected society and is being spread in a multitude of different forms with diverse goals, which we refer to as domains of deception. The most crucial research task in the field of deception is identification of deception, which in most cases involves a machine learning model making the binary classification of Deceptive or Not Deceptive. These classification models are very important as they can help protect the security of an organization by preventing phishing emails from being read, protect online retailers from being flooded with fictitious reviews, and many other tasks depending on the domain of deception they are trained to handle. There has been a fair amount of research focused on the classification of deception, however most research has focused on one domain of deception exclusively. In this work we look at the quality of multiple datasets across different domains of deception, investigate the traces that deception may leave across domains by performing multiple tests using machine learning models, as well as ascertain how using linguistic cues to identify deception performs over multiple domains.**

***Index Terms*—Natural Language Processing, Deception, Data Quality, Machine Learning, Computational Linguistics**

I. Introduction

Deception occurs in a variety of forms, from verbal to textual, and from phishing emails to simply lying to friends. Textual deception is increasingly on the rise in today's internet connected society. With Fake News all over social media like Twitter and Facebook, or Jobs Scams and Phishing emails clogging inboxes, people encounter a multitude of examples of deception every day. As opposed to verbal deception, with textual deception it is feasible to recognize these examples in real time and filter them so they never reach the user.

In this work we train machine learning classification models utilizing datasets from five different domains of deception: Job Scams, Fake News, Product Reviews, Political Statements, and Phishing. This allows us to investigate the domain independence of deception, which is the existence of traces that deception leaves behind that remain consistent across all domains. There are only a handful of works that exist on the domain independence of deception and there are mixed opinions on whether these domain independent traces exist. If a domain independent deception detection model could be created it would increase

978-1-6654-6090-3/22/$31.00 ©2022 IEEE
DOI 10.1109/BDCAT56447.2022.00042

the efficiency of deception detection systems as only one model would be necessary, cutting down on training time and space taken on the computer. Finding a link between separate domains of deception would also make it easier to train detection models on new domains of deception, as they come to light. With new insights into the core of deception, researchers could have a head start on new domains of deception. Working with multiple datasets also allows us to test techniques across multiple domains to determine the efficacy of the technique as a whole, rather than on a specific dataset or domain.

A previous technique that we are particularly interested in testing the efficacy of over multiple domains is using Linguistic Inquiry and Word Count (LIWC) cues to train classification models. Many works have used LIWC to obtain linguistic cues and train machine learning classifiers on those features, however these works do not investigate their performance over multiple datasets. In addition, not much research has been done with the recently released LIWC-22 version [2], which introduces new categories, capabilities, and improvements over previous versions.

In this work we use LIWC-22 to identify best practices for domain independent and domain specific deception detection, our contributions are:

- A review of the quality of our datasets, using NLP techniques.
- Comparisons of many different machine learning feature sets to determine the ideal feature set for each domain.
- Comparisons of multiple different machine learning model architectures to determine the best model when working with LIWC data.
- Experiments involving models trained on multiple different domains and tested on multiple different domains to determine the domain independence of deception.

II. Prior Work

Little work has been done in the field of domain independent deception; however, there are some works that tackle this idea, including a taxonomy for the field, presented in [12]. Others have worked on training classification models using multiple datasets such as in [15] and [10].

Despite the little work on domain independent deception there is a strong foundation for the techniques presented in this paper, when used in single domain deception detection. Linguistic cues have been used in the past to classify deception from many domains including Fake News [11], phishing [3], [5], [13], interview dialogues [7], and conversational dialogues [4].

We hope to expand on the work done on domain independent deception using techniques that have been proven in other domains of deception.

III. Datasets

We employ deception classification methods on five different domains of deception, Fake News, Job Scams, Political Statements, Phishing, and Product Reviews. The datasets used for each of the domains, as well as the process for cleaning them, are explained in detail in the work [15]. For convenience we include a table which includes the number of deceptive and non-deceptive entries that are in each dataset, Table I.

TABLE I: The number of deceptive and non-deceptive entries in each dataset

Dataset	Deceptive Entries	Non Deceptive Entries
Job Scams	608	13,735
Fake News	27,486	34,615
Product Reviews	10,493	10,481
Political Statements	5,669	7,167
Phishing	6,134	9,202

IV. Data Quality

Before discussing further, it is important to take a deeper look at the datasets we are working with to determine any factors that may influence the results [14]. One technique used to dig deeper was to compute the average token length for deceptive and non-deceptive documents in every dataset. These results are shown in Table III. There are a few interesting results to note. For example, the Job Scams dataset has a higher token length than any other dataset. After further investigation nothing out of the ordinary was detected. It may simply be that due to the nature of the domain being more rehearsed and revised than others Job Scams uses bigger words on average, because they have the capabilities to do so. Also, the Phishing dataset has a larger gap between deceptive and legitimate token length than other datasets, as well a standard deviation of token length that is greater than the average token length. This large standard deviation seems to imply the dataset contains links, email addresses, or phone numbers that drive the average token length up due to being much larger than the average word. These large tokens may be especially present in deceptive examples as the average token length is greater for those texts. Further investigation provides more weight to this observation as many deceptive texts in the phishing datasets contain emails, often times multiple, such as “plasma_tv_2003@earthlink.net”.

Also of note are the average number of tokens per document. In this category Political Statements and Product reviews have relatively few tokens per document, which may make them harder tasks for classification. Also, Phishing has a very large standard deviation for number of tokens per document, which may mean the dataset is a bit erratic.

In addition to token lengths, Inverse Document Frequency (IDF) scores for each dataset were also computed, which details the words that appeared most and least frequently in each dataset. The IDF score results are reported in Table II with the bottom 30 words (not including stopwords) by IDF score from each dataset (the most commonly occurring words) being shown. IDF scores were computerd per token, with the tokenization being done through spaCy, which is why some terms such as ”don” or ”000” are present, as they were separated by punctuation from their full versions of ”don’t” or ”100,000”. There were a few pertinent observations from this data.

For the Fake News and Political Statements datasets, the words that were most prevalent were also heavily time dependent. For example, in the Fake News dataset the deceptive examples frequently mention words such as *2016*, *hillary*, and *clinton*, which suggest a heavy concentration centering around the 2016 presidential election. The Political Statements dataset also uses those time sensitive words such as *obamacare*, *law*, and *clinton*, appearing more frequently in deceptive examples than legitimate ones. This suggests that these datasets are temporally qualified and may not contain information from other time periods of Fake News or Political Statements, which could mean that models trained using this data could not work as well on these domains as they exist in our world today.

As for the other datasets the Phishing

TABLE II: The bottom 30 words by IDF score in each dataset. Words in bold are words that were found in the bottom 30 for both Legitimate and Deceptive Examples.

Dataset	Bottom 30
Job Scams Legitimate	**team**, **work**, **looking**, **new**, **experience**, **company**, **business**, **development**,**responsibilities**, **working**, **management**, time, **support**, **customer**, based, **position**, role, **provide**, **service**, **skills**, product, help, **job**,**responsible**, opportunity, **including**, **services**, design, high, environment
Job Scams Deceptive	**work**, **company**, **experience**, time, **team**, **responsibilities**, **job**, **looking**, **position**, **business**, required, **management**, **support**, **service**, **services**, **customer**, information, **skills**, **provide**, **new**, duties, **responsible**, **including**, requirements, **development**, systems, products, process, **working**, ensure
Fake News Legitimate	**said**, **president**, **new**, **people**, **trump**, **year**, told, **state**, **donald**, **states**, **time**, government, united, **years**, house, **did**, including, **like**, **country**, **just**, week, republican, national, **say**, called, **make**, **news**, political, campaign, white
Fake News Deceptive	**said**, **people**, **just**, **trump**, **like**, **president**, **time**, **donald**, **new**, right, know, **make**, way, **news**, don, going, 2016, **did**, **state**, **years**, american, clinton, hillary, **year**, **say**, **states**, america, **country**, think, world
Phishing Legitimate	**sent**, subject, time, pm, **email**, know, just, **new**, like, **message**, **com**, 10, national, wrote, make, need, 2016, thanks, org, **information**, don, **mail**, 11, want, work, today, good, 12, think, use
Phishing Deceptive	account, click, dear, **information**, **email**, thank, link, **mail**, security, service, customer, address, access, online, help, receive, **message**, **sent**, rights, reply, reserved, **new**, update, member, user, protect, using, bank, copyright, **com**
Political Statements Legitimate	**percent**, **state**, **000**, **years**, **year**, **tax**, **obama**, **states**, **people**, **million**, **health**, **jobs**, **president**, **new**, **texas**, country, **federal**, **care**, **billion**, **taxes**, **budget**, united, **said**, time, **voted**, rate, american, 10, americans, **government**
Political Statements Deceptive	**obama**, **percent**, **president**, **state**, **health**, **care**, **tax**, **people**, **000**, **years**, **year**, barack, **states**, **new**, **government**, **voted**, **million**, **said**, **jobs**, **billion**, law, **federal**, wisconsin, plan, **budget**, obamacare, **texas**, **taxes**, clinton, money
Product Reviews Legitimate	**great**, **like**, just, **good**, **use**, **time**, **love**, **really**, **little**, **product**, **price**, **don**, **bought**, **easy**, **quality**, **nice**, **used**, **work**, **does**, **works**, **recommend**, **buy**, **got**, **better**, ve, make, need, did, **perfect**, fit
Product Reviews Deceptive	**great**, **good**, **like**, **really**, just, **product**, **use**, **quality**, **love**, **time**, **price**, **easy**, **bought**, **recommend**, **got**, **don**, **nice**, **buy**, **works**, **little**, **work**, **used**, **perfect**, **better**, using, best, look, looks, **does**, looking

dataset had the most striking difference between deceptive and non-deceptive texts, with each having an almost completely different 30 most common words. The legitimate emails were reminiscent of work emails, including words like *want*, *message*, *work*, *need*, and *make*. Whereas the deceptive emails were more reminiscent of tech support emails with words such as, *account*, *click*, *link*, *security*, *service*, *access*, *online*. This may mean that phishing emails are easier to classify based on subject matter than other datasets or it may be a result of how the dataset was collected, from various sources, which may show a greater difference in subject matter.

TABLE III: Average and standard deviation token length for each dataset. Results reported as Average (±Standard Deviation)

Dataset	Average Length	Average Deceptive Length	Average Legitimate Length	Average Number of Tokens per Document
Job Scams	5.659 (±3.227)	5.816 (±3.304)	5.652 (±3.224)	186.8 (±130.1)
Fake News	4.937 (±2.900)	4.905 (±3.072)	4.959 (±2.774)	558.6 (±618.5)
Phishing	4.862 (±5.058)	5.366 (±5.454)	4.728 (±4.939)	404.5 (±1017.5)
Political Statements	4.996 (±2.660)	5.036 (±2.681)	4.965 (±2.644)	17.7 (±7.7)
Product Reviews	4.269 (±2.522)	4.275 (±2.628)	4.264 (±2.441	70.2 (±87.1))

These dataset quality observations are something to note but they should not stop us from utilizing them in this work, as these characteristics are likely reflective of the real world and could help in classification. However, it may be useful in future work to test these observations and their effect on classification, using Fake News datasets from a single time period to train classification models, then testing them on datasets from other time periods to test the temporal aspect of Fake News, or investigating the difference in subject matter of Phishing emails.

V. Linguistic Features

LIWC is a software that extracts a variety of linguistic cues from any given text. As of LIWC-22 there are over 110 different categories of linguistic cues with over 20 of them being new in this version. These categories range from the tone of the text, to words that involve social status, to words that show anger, and so on. Within LIWC there are multiple dictionaries of words, one for each category. Each dictionary contains words that pertain to the category, and the software will use these words to determine what percentage of the given text belongs to that category. The output of LIWC are these percentages for each category, as well as Word Count and Words per Sentence.

Once we collected these statistics for every dataset, we conducted statistical tests to determine which LIWC categories would be significant indicators of deception. We tested for each dataset to determine if a category of words occurred at different rates in deceptive and legitimate texts. We used a t-test with unequal variances assumption, and to control for family wise error rate we used the Holm-Bonferroni method [1].

The number of categories that showed significant differences between legitimate and deceptive texts in each dataset ranged from 38 to 112. Eleven categories were significant in all five datasets, however none of those 11 had the same direction in all datasets. This means that while they were significant in all five datasets, they were found more commonly in deceptive texts for some datasets and found more commonly in legitimate texts for other datasets. For example, the tone_pos category was found more often in deceptive texts for four of the five datasets but was found more commonly in legitimate text in the remaining dataset.

Now that we had this information, we could then determine what subset of these categories is the best when used for machine learning deception classification. We initially tested six different feature sets and added three later. The initial 6 feature sets tested were categories significant in all 5 datasets (5 significant), categories significant in 4 or more datasets (4+ significant), categories significant in 3 or more dataset

TABLE IV: F-score of each dataset for each feature set in the form of Mean(±Standard Deviation)

Dataset	5 Significant	4+ Significant	3+ Significant	2+ Significant	1+ Significant
Job Scams	95 (±0.0)	**96** (±0.6)	**96** (±0.6)	**96** (±0.6)	**96** (±0.6)
Fake News	77 (±0.6)	88 (±0.0)	90 (±0.6)	90 (±0.6)	**91** (±0.6)
Product Reviews	59 (±0.6)	63 (±1.0)	**64** (±0.6)	**64** (±1.2)	**64** (±0.6)
Political Statements	56 (±0.6)	57 (±0.6)	58 (±0.0)	**59** (±0.6)	58 (±0.6)
Phishing	90 (±0.6)	94 (±0.6)	**95** (±0.0)	**95** (±0.6)	**95** (±0.6)

TABLE V: Continuation of the Table IV

Dataset	Individual	Top 5	Top 10	Top 20
Job Scams	**96** (±0.6)	95 (±0.0)	95 (±0.0)	95 (±0.6)
Fake News	88 (±0.6)	64 (±0.6)	71 (±0.6)	83 (±0.6)
Product Reviews	62 (±1.0)	55 (±0.6)	58 (±0.6)	60 (±0.6))
Political Statements	55 (±0.6)	51 (±0.0)	53 (±0.5)	54 (±1.5)
Phishing	94 (±0.6)	85 (±1.0)	89 (±0.6)	92 (±0.6)

(3+ significant), categories significant in 2 or more datasets (2+ significant), categories significant in 1 or more datasets (1+ significant), and a feature set where each dataset used only the categories found significant in that dataset (Individual). After these tests, we obtained feature importance values for the models using the 3 or more significant feature set, then for each dataset we took the top 5, 10, and 20 features and used those as their own feature set. All results were obtained using a Random Forest Classification model with an 80-20 train-test split and were averaged over three trials, results are shown in Tables IV, and Table V, with average F-score and standard deviation shown. All Random Forest models in this work were created using base specifications in the python library sklearn (version 1.1.2) [8].

The highest F-scores from these experiments are shown in bold and are mostly from the feature sets 3+ significant, 2+ significant, and 1+ significant. Our “best” results from these experiments are from the 3+ significant feature set, as the gain in performance is minor when adding more features to obtain the 2+ significant and 1+ significant features. Henceforth in this work we will use the feature set 3+ significant for deception detection.

This result is noteworthy because the more general approach, using features deemed significant in a majority of the datasets, performed better than using feature sets tailored to each dataset. The Individual feature set contains only LIWC categories that were found significant in that specific dataset, however for all datasets it performs worse than the 3+ significant feature set, except for the Job Scams dataset where they perform equally well. This difference in performance is also not due to an increased number of features being used, as the 3+ significant feature set contains 87 different features whereas the Individual feature sets contain anywhere from 38 to 112. This suggests that a more general and broad approach to deception detection could be more useful in the long run than specific models tailored to each domain. This also lends some evidence to the existence of domain independent traces of deception, where the domain independent feature set may have picked up on the nature of deception itself rather than within the specific domain, as the Individual models did.

We have reported F-scores from Random Forest models because those models generally performed the best over every feature set. We tested 6 different models on every feature set, PART (through WEKA [6]), Support Vector Machines (SVM) with a linear kernel, SVMs with a Radial Basis Function (RBF) kernel, SVMs with a degree 3 kernel, Multilayer Perceptrons (MLP), as well as Random Forest. For all feature sets Random Forest models performed the best, with MLPs not far behind for all datasets. Thus, we determined that Random Forest models were the best equipped to perform deception detection when using LIWC statistics. Deception detection is unique when using LIWC statistics because the model is never given the text as an input. The model only receives LIWC statistics, an abstraction of the text, as input. This may change the ideal model to tackle this task, which is why this is something we looked into and performed tests to conclude.

VI. Methodology

To evaluate the effectiveness of our techniques and the domain independence of deception we conducted multiple experiments using different models, as well as different train/test sets. All results are 5-fold cross validated and in an effort to create fair comparisons across different models the same folds of each dataset were used across all experiments. Also, in these experiments when we refer to LIWC Linguistic Features we refer to the 3+ significant feature set only, as that feature set was determined to be the most effective.

VII. Results

A. Single Dataset Baselines

To fairly compare the models and experiments, we must first establish a baseline. We do this through single dataset models, models trained and tested on one single dataset. Besides the dataset, nothing else was changed about the model for each data point.

TABLE VI: F-score and MCC for Random Forest Classification using LIWC Linguistic Features only

Dataset	F-Score	MCC
Job Scams	0.951	0.373
Fake News	0.894	0.785
Product Reviews	0.643	0.288
Political Statements	0.575	0.142
Phishing	0.943	0.881

F-score and Matthew's Correlation Coefficient (MCC), which is an adaptation of Pearson's Correlation Coefficient that helps summarize the confusion matrix in a single value, results for these baselines are found in Tables VI,

Looking at the results in Table VI, for Random Forest models using only LIWC features, there are 2 domains where the models perform well, that being Fake News and Phishing, where the models show F scores in or close to the .90s as well as MCC scores above .7, which is very good. In addition, there are 2 datasets where these models perform fairly poorly in comparison, showcasing results barely above random with the Product Reviews and Political Statements datasets. It is interesting to note that these two datasets have the lowest average number of tokens per file, which may make them harder to classify with this method, as there are fewer linguistic features to extract. Finally, there is one dataset for which these models give a confusing report where the F-score is quite high, but the MCC is fairly middling. This is the Job Scams dataset, and this is likely a result of the Jobs Scams dataset being heavily unbalanced, so the model learns to classify most of everything as the dominant type, which in this case in non-deceptive, which leads to high F-scores but a fairly off confusion matrix, leading to a lower MCC score than some of our other high performing datasets. This heavily unbalanced dataset does prove a challenge for machine learning models, as the model may learn some unwanted behavior, but it is also more true to the real world, as real job offers heavily outweigh the number of scams offers that exist. Imbalanced learning discussions are outside of the scope of this work, and it has been discussed at length in other research such as [9]

B. Cross dataset Experiments

We conducted many cross-dataset experiments to determine the effect different datasets have on model results, as well as the domain independence of deception. The experiments are as follows: Train on All Test on All, where the model is trained using training data from all of the datasets, then tested on each dataset individually, Train on One Test on All, Where the model is trained using data from only one dataset, and then tested on all datasets individually, and finally Train on All Minus One, where the model is trained using the data from 4 out of 5 of the datasets, then tested on each dataset (including the removed dataset) to determine the effects of removing those datasets from the training set.

Looking at the results from the models that only had access to the LIWC features, their Train on All Test on All results are located in Table IX. In this we can see that the results are not terribly different from the models trained using only one dataset. The biggest drop off in F-score was Phishing, decreasing by 0.038, which is substantial, but the performance is still good at .905 F-score. The MCC also stays at around the same value in the new models. This shows that it is possible to create a model that does fairly well in all of the different domains of deception, which implies that there is some sort of common thread between them, as the tasks are not so different that the model cannot perform reasonably.

The results for LIWC features only Train on One Test on All are in Table VII. Results shown in italics are the same results shown in the single dataset baselines. These results show that mostly the training from one dataset is not very transferable to other datasets, as the models perform generally poorly. The best performing model was the model trained on political statements which was able to obtain an impressive 0.799 F-score on the Job Scams dataset, as well as an above random F-Score of 0.567 on the Phishing dataset. The lowest performing dataset was the Fake News dataset. The model trained on Fake News data did very poorly on other datasets with significantly lower than 0.5 results. This may have happened because the Fake News dataset was our biggest dataset, and the model fit itself very closely to the dataset, with not a lot of room for error.

The next experiment was Train All Minus one, and the results for LIWC features only can be viewed in Table VIII. Bold results indicate they were the best F-score for that test set, and the left column is the dataset that was not included in the training set. Predictably, the test set with the worst F-Score was always the set that was not included in the training. An interesting result to note is that the best results were most of the time achieved when the Fake News dataset was left out of the training set. This may once again be due to the fact that the Fake News dataset was our biggest dataset, so the model may have fit a little too closely with the fake news dataset in comparison to the other datasets.

TABLE VII: F-score and MCC for “Train on One Test on All” scenario using Random Forest Classifier and LIWC features only.

		Test Set				
		Jobs	News	Prod	Poli	Phish
Train set	Jobs	*0.951*	0.401	0.334	0.406	0.451
	News	0.179	*0.894*	0.339	0.317	0.397
	Prod	0.612	0.457	*0.643*	0.481	0.573
	Poli	0.799	0.491	0.500	*0.575*	0.567
	Phish	0.693	0.422	0.362	0.452	*0.905*

TABLE VIII: F-score and MCC for “Train on All Minus One” scenario using Random Forest Classifier and LIWC features only.

		Test Set				
		Jobs	News	Prod	Poli	Phish
Left Out Set	Jobs	0.558	0.874	0.622	0.563	0.907
	News	**0.950**	0.422	**0.631**	0.554	**0.927**
	Prod	0.947	0.869	0.515	**0.568**	0.905
	Poli	0.949	0.866	0.616	0.528	0.907
	Phish	0.948	**0.877**	0.614	0.566	0.404

TABLE IX: F-score and MCC for “Train on All Test on All” scenario using Random Forest Classifier and LIWC features only.

Test Dataset	F-Score	MCC
Job Scams	0.948	0.287
Fake News	0.864	0.734
Product Reviews	0.622	0.250
Political Statements	0.560	0.117
Phishing	0.905	0.802

VIII. Discussion

The best way to put these results into context, to be able draw meaningful conclusions from them is to compare them to the results obtained in [15], which performs similar experiments on the same datasets, using a BERT model. In that work they present impressive results with comparable, yet superior F-Scores for classification. The greatest difference in F-scores in shown in Fake News where they report a F-score of .9973, which is a difference of .1033, and the smallest difference is in the Political Statements dataset, where they report a F-score of .5940, for a difference of -.019 when compared to the F-score from our single dataset baseline model. The average F-score for our single dataset baseline models is .7936, and for their models the average is .8596. Their work does exhibit similar trends however, with Fake News, Phishing, and Job Scams having the best F-scores and Product Reviews, and Political Statements further behind. As well as looking at their experiments with models trained on one dataset tested on all datasets, when testing on domains the model has not been trained on the model has an average F-score of .591465, whereas the same average for our models is .46165. This difference is much larger than the difference between the average of the single dataset baselines so we can conclude that BERT is better on domains that it has not encountered before, and this makes sense because BERT has a much larger knowledge of the language itself as opposed to being a model trained from beginning to end on a set of LIWC categories from one dataset.

TABLE X: The top 5 most important features from each dataset

Dataset	Top 10 Categories
Fake News	focuspresent, quantity, i, lack, negate
Job Scams	tentat, Clout, Linguistic, allnone, ppron
Phishing	Authentic, polite, Drives, number, Apostro
Political Statements	certitude, det, affiliation, tone_pos, auditory
Product Reviews	Linguistic, Apostro, motion, tone_pos, Authentic

While results from our models fall behind those of BERT models, these models do have many advantages over BERT models. One of the biggest advantages of models that utilize LIWC categories is that they are inherently much more explainable than BERT models. Because the inputs are numerical, they can be varied to ascertain the effect on the output, meaning feature importances can easily be calculated. Then when these feature importances are calculated all LIWC categories mean something and can be connected to the language and the real world. For example, results obtained from computing feature importances on some preliminary models from this project are show in Table X. These models were trained and tested on a single dataset over three trials using an 80-20 train-test split. These results provide actionable insights based on the model. From these one can conclude that how positive the tone is plays a role in deception in the Political Statements and Product Reviews datasets, as well as many other observations that include data not listed here. This is helpful for research that wishes to characterize deception in a specific domain or across multiple domains. Another benefit of these models is that they are much more lightweight than BERT models, they train much faster and take up less space on a computer which is useful if models like these are deployed in the field.

IX. Conclusion

Some researchers think that a disadvantage of domain-independent deception detection is that domain specific features cannot be exploited, such as URLs and emails in the case of phishing or company names in the case of job scams. However, that is not necessarily true, since one can form ensembles of a domain-independent deception detector with domain-specific features. Of course, these techniques require domain expertise and knowledge, and possibly also additional time when training these models. With domain independent deception training new models, especially in new domains, becomes much easier. Specific knowledge of the domain also becomes less important.

When looking at the domain independence of deception the results from this work are promising yet inconclusive. There is promise that a domain independent thread of deception exists, as shown in the results of the models trained using different feature sets. In those models better F-scores were achieved when using a feature set that was more generic for all datasets, than when the feature set was tailored to each dataset individually. This shows that it is better to train the classifier to classify deception than it is to train it to classify deception on the Job Scams dataset. However, the models in this work did not show a great degree of generalizability to other domains when trained on only one. This suggests that there may be some set of LIWC features that is optimal for all deception *if we look at multiple possible domains* *of deception*, but the one shown here is an approximation of that and does not always work the best across different datasets not included in the training. Future work in this area may do just that, extending these methods to many more domains of deception to ascertain what may be domain independent traces of deception, and characterize deception using LIWC categories so that humans can gain more insight about what deception really looks like.

Acknowledgments

We thank the reviewers for their constructive comments. This research was supported by NSF grant 1950297. Verma's research was also partially supported by NSF grants 1433817 and 2210198, ARO grant W911NF-20-1-0254, and ONR award N00014-19-S-F009. He is the founder of Everest Cyber Security and Analytics, Inc.

References

[1] Hervé Abdi. Holm's sequential bonferroni procedure. *Encyclopedia of research design*, 1(8):1–8, 2010.

[2] Ryan L Boyd, Ashwini Ashokkumar, Sarah Seraj, and James W Pennebaker. The development and psychometric properties of liwc-22. *Austin, TX: University of Texas at Austin*, 2022.

[3] Avisha Das, Shahryar Baki, Ayman El Aassal, Rakesh Verma, and Arthur Dunbar. Sok: a comprehensive reexamination of phishing research from the security perspective. *IEEE Communications Surveys & Tutorials*, 22(1):671–708, 2019.

[4] Nicholas D Duran, Scott A Crossley, Charles E Hall, Philip M McCarthy, and Danielle S McNamara. Expanding a catalogue of deceptive linguistic features with nlp technologies. In *FLAIRS Conference*, 2009.

[5] Ayman El Aassal, Shahryar Baki, Avisha Das, and Rakesh M Verma. An in-depth benchmarking and evaluation of phishing detection research for security needs. *IEEE Access*, 8:22170–22192, 2020.

[6] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[7] Sarah Ita Levitan, Angel Maredia, and Julia Hirschberg. Linguistic cues to deception and perceived deception in interview dialogues. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1941–1950, 2018.

[8] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[9] Fatima Zahra Qachfar, Rakesh M Verma, and Arjun Mukherjee. Leveraging synthetic data and pu learning for phishing email detection. In *Proceedings of the Twelveth ACM Conference on Data and Application Security and Privacy*, pages 29–40, 2022.

[10] Rodrigo Rill-García, Luis Villasenor-Pineda, Verónica Reyes-Meza, and Hugo Jair Escalante. From text to speech: A multimodal cross-domain approach for deception detection. In *International Conference on Pattern Recognition*, pages 164–177. Springer, 2018.

[11] Rui Sousa-Silva. Fighting the fake: A forensic linguistic analysis to fake news detection. *International Journal for the Semiotics of Law-Revue internationale de Sémiotique juridique*, pages 1–25, 2022.

[12] Rakesh M. Verma, Nachum Dershowitz, Victor Zeng, and Xuting Liu. Domain-independent deception: Definition, taxonomy and the linguistic cues debate. *arxiv.org*, 2022.

[13] Rakesh M Verma and David J Marchette. *Cybersecurity analytics*. CRC Press, 2019.

[14] Rakesh M Verma, Victor Zeng, and Houtan Faridi. Data quality for security challenges: Case studies of phishing, malware and intrusion detection datasets. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 2605–2607, 2019.

[15] Victor Zeng, Xuting Liu, and Rakesh M. Verma. Does deception leave a content independent stylistic trace? In *Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy*, CODASPY '22, page 349–351, New York, NY, USA, 2022. Association for Computing Machinery.