Semantic Analysis and Classification of Twitter Posts

George Darkens
gd17659
University of Essex
Colchester, Essex
gd17659@essex.ac.uk

***Abstract*—The massive explosion of readily accessible raw language data stemming from the growth of the World Wide Web has led to great advances in natural language processing. With the ever expanding source of raw language data this provides, many researchers have focused on using this enormous new corpus to train and improve their systems. However, some areas of the web are more complex for these systems to understand, such as social media. Much of the language from websites such as Twitter are atypical compared to the formal language that processors usually operate on, partially due to tight character constraints. This language is far denser, noisier, conversational, and idiosyncratic compared to the corpus' that systems would typically learn from and operate over, as well as the unique characteristics that language on the Web has.**

This project seeks to build a language model that can accurately; classify Twitter posts that contain hate speech, determine a posts sentiment, and predict which emoji is most likely to be included given the provided context. Building upon a natural language processor and training it on a benchmarking corpus, this project seeks to beat competing models as evaluated by a standardised benchmarking system for these given tasks.

***Index Terms*—Natural Language Processing, Social Media, Semantic Analysis, Twitter, Language Classification**

I. Introduction

Natural language processors (NLPs) are systems created to process, analyse, and 'understand' human language. Their applications are vast; originally used to translate single sentences of text from Ru ssian to English in 1954 [1], but now with modern technology and expanded research has expanded to tasks such as accurate natural language generation and speech recognition. Models such as GPT created by T. Brown et al. at OpenAI displays the incredible leap that NLPs have taken when combined with modern techniques. GPT-3, the latest version, is trained on a corpus so diverse it can code in various programming languages, create guitar tabs, and even generate text such as news articles that are indistinguishable from that created by a human. [2] [3]

The advances a system like GPT can create over so few years and generations of the system is the result of the exponentially increasing amount of freely accessible raw data produced by the World Wide Web. GPT has trained on billions of pieces of data, or 'tokens' - the percentage of those that were not sourced from the Web is only 16%, these being from books. [3, Fig. 2.2]

Of the subcorpora that the Web can be roughly divided into, social media is one that is inherently difficult for NLPs to process and analyse. Even when all tweets are written in a single language, the nature of social media, and sites such as Twitter specifically, make the natural language used far removed from formal natural language that models had trained upon prior. Twitter forces users to compress whatever they seek to post down to at most 240 characters, promotes inter-user conversations, and fosters atypical language features like hashtags and emojis. Twitter grants NLPs less contextual clues to build their models atop and interpret from, and those that are there are far more idiosyncratic than their typical corpora.

These features of Twitter, as well as the gradual development of an Internet dialect, combine to create a unique corpus for research and analysis. *TweetEval* is a standardised evaluation protocol created by F. Barbieri et al. to assess and benchmark the accuracy of a model in multiple heterogeneous tasks. Of these, the author will be creating a system targeting three; hate speech detection, sentiment analysis, and emoji prediction.

Creating a LM with the goal of analysing sentiment is one that is increasingly popular, especially for companies, researchers, and political parties who want to know the unfiltered and live opinions of Web users on certain topics.

This author seeks to create a language model (LM) capable of determining the semantics of Twitter tweets for the use of detecting and determining hate speech, the sentiment of a post, and the accurate prediction of the placement or applicability of emojis. This model will be compared using the evaluation protocol laid out by [4], utilising the training and validation datasets it provides, and tested against their test set for an unbiased evaluation of a final model.

II. Literature review

The *TweetEval* paper [4] defines and presents both the testing methodology and relevant datasets which the author will utilise to create this projects language model (LM). It discusses the unique challenges presented by Twitter as it pertains to an NLPs ability to accurately interpret, classify, and understand the unique 'dialect'

used. While it points out modern NLPs lacking accuracy in this area, it uses the benchmarking framework it defines to compare the results of three differently trained categories of NLP. The first is a pre-trained model, the second only pre-trained on Twitter data, and a third combining both, being pre-trained prior to additional training on Twitter data.

The results of this analysis concluded that "...using a pre-trained LM may be sufficient, but can improve if topped with extra-training on in-domain data." [4, Ch.5, p.5] They suggested that pre-training the LM prior to training it on Twitter was advantageous since Twitter does contain both noisy and formal text. In fact, they point to a study by Hu et al. [5, Ch.5, p.252] suggesting that Twitter is in fact "...markedly more standard and formal than SMS and online chat, closer to email and blogs, and less so than newspapers."

While Hu et al. suggest Twitter has greater formality than thought, they also discuss the features of language on the site. They concluded that Twitter users are "linguistically unique" compared to other mediums, both online and not. An example presented is the use of both first-person and third-person pronouns, where other mediums would typically use just one.

Alexander Pak and Patrick Paroubek (2010) [6] discuss how techniques tuned to Twitter's unique use of internet dialect. Their training corpus of 300,000 tweets was derived from queries to Twitters API for tweets containing "happy emoticons" and "sad emoticons". Since this paper was from 2010, they assume that "...an emoticon within a message represents an emotion for the whole message and all the words of the message are related to this emotion." [6, p.1321] Whilst this may have been more reasonable at the time, since this papers publication in 2010 the maximum length of a tweet has doubled, from 120 to 240 characters. However information discussed by Twitter employees on the length of tweets since then suggests that only 5% of tweets exceed 190 characters, and that abbreviations and text speak have declined significantly in some cases, creating a less noisy and more formal corpus. Twitter saw that '...abbreviations like "gr8" is down by 36%, use of "b4" is down by 13%," and "sry" has dropped 5%. Other words have increased as result, including "great" (+32%), "before" (+70%) and "sorry" (+31%).' [7]

The sentiment analysis presented in [6, p.1321] shows aspects that the author will be including in this projects solution. An obvious aspect is the papers use of 'emoticons' to gauge opinion and emotion. Whilst this project won't specifically be using emoticons (e.g. :), :(, ...) it will be using emojis (e.g. 😊, 😢) - the difference being one is comprised of characters, and the other an image. Not only can emojis be used to derive a stance [8], they could also be used to in part to classify hate speech, although not as the only variable. This project will also seek to predict which emoji a user would include in their tweet using only their text. A possible barrier to this presented in [8, Ch.4.2, p.47] is added noise from emojis being used in a sarcastic manner, as well as mixed sentiments being presented by emoticons that display opposite emotions.

III. Methodology

One of the metrics used to directly evaluate the performance of this project's solution will be those defined by F. Barbieri et al. in their benchmark paper [4]. The paper states the use of a macro-averaged F1 over all mapping classes. F-scores are calculated using the precision and recall of each model, where the precision is the fraction of tweets accurately tagged, and the recall is the fraction accurate tags. These calculations are used to evaluate all datasets apart from in stance detection, irony detection, and sentiment analysis.

This authors project will use the sentiment analysis dataset, and therefore the evaluation of this will be calculated with a macro-averaged of the total recall the model produces. In the case of sentiment analysis, the recall value is the fraction of tweets that are accurately tagged compared to their labeling.

The results of [4] indicate that LMs that score the highest F-scores are pre-trained prior to being specifically trained upon a Twitter corpus. As such, this project will be utilising a pre-existing NLP, specifically the LM will likely use aspects of the Natural Language Toolkit (NLTK). [9] The NLTK is a Python library that has a diverse range of natural language methods, ranging from stemming and tokenizing, to part-of-speech (POS) tagging and its own sentiment analysis methods.

Of the three selected datasets, emoji prediction is the most unique of them in terms of the required solution. Whilst not only for use on emoji dataset, the author suggests the use of phrase chunking, especially adjective-phrase and verb/adverb-phrase chunking would assist with deriving emotions from short, possibly single sentence tweets. Phrase chunking will require use of the NLTK's POS tagging functionality to identify words to be nouns, verbs, adjectives, adverbs, etc.

The 'hate' dataset is also unique as it is specific to hate against two communities; women and immigrants. To improve the results gained here, POS tagging could be used for named-entity recognition. This is where named 'entities', such as the groups defined here, are identified and classified. Upon identification, it would likely be easier to determine if hate speech is being used toward these groups or not.

The training, testing, and evaluation datasets being used in this project are provided by *TweetEval*, but are modified versions of challenges set at SemEval events over the past few years. [10] [11] [12]

IV. Results

The author has done preliminary work in loading, processing, and exploring the datasets relevant to the

tasks chosen. All work done throughout this project will be uploaded on the authors GitHub page for this module[1], and preliminary work can be found at `./Assignment1/init_testing_exploratory/`. Initial work on preprocessing such as stemming and stopword removal has been demonstrated, as well as manipulation of the dataset. One issue the author repeatedly found was Python's reluctance to operate with emojis, meaning particular attention must be used in the encoding of data when downloaded, manipulated, or passed through packages or methods native or not to Python. Work at this stage in the project towards a trained ML and testing has not yet begun.

Both at the conclusion, and during this projects evolution, the solution designed will be directly compared using the *TweetEval* benchmarking methodology, and thus will draw direct comparisons to the results obtained in their work. Listed on their GitHub repository is a table of results listed in their paper, as well as a newer LM by Dat Nguyen, Thanh Vu, and Anh Nguyen that surpasses all previous LMs in all categories, bar one. [13]

V. Discussion

The corpus' provided by *TweetEval* are preprocessed to a degree prior to the preliminary results gathering. The *TweetEval* paper discusses how "...the preprocessing pipeline is equal for all tasks: user mentions are obfuscated and line breaks and website links are removed." Reviewing the data in one of the `./tweeteval/datasets/` directory shows that detectable twitter usernames, that is those that are explicitly called by username with the '@' symbol, have been replaced with '@user'. Whilst it would seem many website URLs have been removed from the corpus, some persist. A significant portion of those remaining are automatically generated links inserted by Twitter for users using their URL shortening domain, '`https://t.co/`'. Thus, this projects solution will require it's own noise removal methods to ensure that few URLs or otherwise non-desirable, non-natural language text is parsed through the model.

Additional preprocessing will be required during this project, even if the result is not used in all parts of the system. Most NLPs will have their own symbolic methods for tasks such as tokenisation, stopping, lemmatization, and stemming since these tasks must be carried out in the course of its operation. It is probable that some of the preprocessing done in this project will be handled by the NLPs methods which have been created by the developers for use in their system. This grants advantages such as faster processing methods, reduced chance of processing errors, and means that these common functionalities don't have to be reinvented by the author.

[1]https://github.com/GDarkens/CE888

VI. Plan

A Gantt chart is included in the appendix, VII-A, and details the planned trajectory of this project. Each period on the chart is two days, encompassing the roughly two months until the submission date for the next report.

Not all functionality, methods, or possible techniques for completing this project have been planned, and the author suggests caution in the interpretation the Gantt charts predictions. The author does have prior experience with the NLTK, POS tagging, and chunk phrasing from previous work on information system classification and so does not expect large delays in this area. The processes for the implementation and integration of these methods into a ML are where a significant portion of time will be spent, as well as training and benchmarking the project.

References

[1] J. Hutchins, "The first public demonstration of machine translation: the georgetown-ibm system, 7th january 1954," 01 2004.

[2] A. Sabeti, "Gpt-3: An ai that's eerily good at writing almost anything," Jul 2020. [Online]. Available: https://arr.am/2020/07/09/gpt-3-an-ai-thats-eerily-good-at-writing-almost-anything/

[3] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[4] F. Barbieri, J. Camacho-Collados, L. Espinosa-Anke, and L. Neves, "TweetEval:Unified Benchmark and Comparative Evaluation for Tweet Classification," in *Proceedings of Findings of EMNLP*, 2020.

[5] Y. Hu, K. Talamadupula, and S. Kambhampati, "Dude, srsly?: The surprisingly formal nature of twitter's language," *Proceedings of the 7th International Conference on Weblogs and Social Media, ICWSM 2013*, pp. 244–253, 01 2013.

[6] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association (ELRA), May 2010. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2010/pdf/385*Paper.pdf*

[7] S. Perez, "Twitter's doubling of character count from 140 to 280 had little impact on length of tweets," Oct 2018. [Online]. Available: https://techcrunch.com/2018/10/30/twitters-doubling-of-character-count-from-140-to-280-had-little-impact-on-length-of-tweets/

[8] J. Read, "Using emoticons to reduce dependency in machine learning techniques for sentiment classification," in *Proceedings of the ACL Student Research Workshop*, ser. ACLstudent '05. USA: Association for Computational Linguistics, 2005, p. 43–48.

[9] T. NLTK, "Natural language toolkit," Jul 2014. [Online]. Available: https://www.nltk.org/team.html

[10] F. Barbieri, J. Camacho-Collados, F. Ronzano, L. Espinosa-Anke, M. Ballesteros, V. Basile, V. Patti, and H. Saggion, "Semeval 2018 task 2: Multilingual emoji prediction," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, 2018, pp. 24–33.

[11] S. Rosenthal, N. Farra, and P. Nakov, "Semeval-2017 task 4: Sentiment analysis in twitter," in *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, 2017, pp. 502–518.

[12] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. Rangel Pardo, P. Rosso, and M. Sanguinetti, "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*. Minneapolis, Minnesota, USA: Association

for Computational Linguistics, 2019, pp. 54–63. [Online]. Available: https://www.aclweb.org/anthology/S19-2007
[13] D. Q. Nguyen, T. Vu, and A. T. Nguyen, "Bertweet: A pre-trained language model for english tweets," 2020.

VII. Appendix

A. *Gantt Chart*