Automatically Finding Undiagnosed Diseases Cases on Twitter

Final Report
11 May 2020

Abstract

While in today's modern healthcare system many patients are correctly diagnosed and treated for their illness, there are others who go without diagnosis. The Undiagnosed Disease Network stands as a resource for these patients, allowing for applicants to enter a pool for extended examination and better treatment options. However, most do not know of this network, and instead post on social media to broadcast their concerns. In this work, we train a feature-based classifier of social media posts to discern whether or not they contain an undiagnosed disease in order to advertise the UDN. We train and evaluate our system on both manually-annotated Reddit posts and Twitter tweets. An extensive ablation analysis outlines the importance of selected features generated from large pretrained models.

1 Introduction

Every year hundreds of men, women, and children face uncertainty when healthcare providers are unable to discover the cause for their symptoms due to the limitation of current clinical knowledge[9]. Diagnosis at the edges of human knowledge calls upon clinicians to be data driven, cross-disciplinary, and collaborative in unprecedented ways. Exact disease recognition, an element of the concept of precision in medicine, requires new infrastructure that spans geography, institutional boundaries, and the divide between clinical care and research[6]. However, the first step in this effort is to identify those who have been afflicted with unknown diseases, and to refer them to a network which can record and potentially diagnose their disease. In this project, we tackle the problem of using social media to identify those who suffer from undiagnosed disease, in order to refer them to the Undiagnosed Diseases Network.

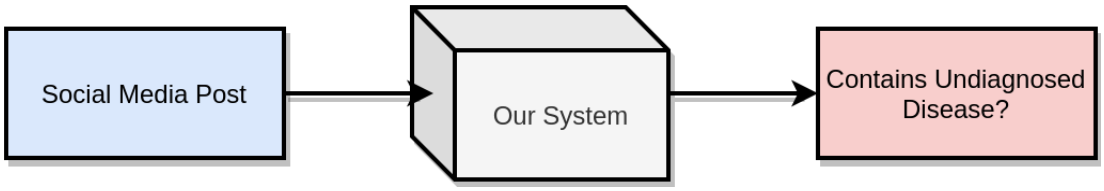

Figure 1: Our system is trained to classify whether a social media submission (such as on Reddit) contains an undiagnosed disease. This system could then be used to alert users of the Undiagnosed Diseases Network.

1.1 The Undiagnosed Diseases Network

The Undiagnosed Diseases Network (UDN) is a research study backed by the National Institutes of Health Common Fund that seeks to provide answers for patients and families affected by these mysterious conditions. Its purpose is to bring together clinical and research experts from across the United States to solve the most challenging medical mysteries using advanced technologies. The UDN, builds on the successes of the Undiagnosed Diseases Program at the National Institutes of Health (NIH UDP), through support from the NIH Common Fund, a coordinating center, six additional clinical sites, and two sequencing cores comprise the UDN [6]. The success of the UDN depends on the collection and subsequent sharing of well-curated clinical and research data both within and outside of the network. It is reported that 4369 applications have been received through the UDN, and 1603 participants have been accepted by the fund[9]. Although the UDN has achieved considerable success, there are some potential issues in the UDN program. One of the issues is that the UDN is not a prevalent website, and it is believed that there are still many more patients with undiagnosed diseases who have not accessed the UDN. Finding more people with such situation and alerting them of the UDN would be of great benefit. In this project, we train a classifier to detect UDN applicants through social media and evaluate our system on ranking tasks and show positive results in a limited data domain [1].

2 Related Work

Given the problem statement above, we wish to perform categorical classification on a textual input. Given the data sparcity of this task, it is desirable to use pretrained models which can be fine-tuned for this task. We review popular pretrained models here.

Word2Vec [3] is a method for automatically learning dense representations of words or tokens within a corpus of text. These representations are vectors of real-valued elements, which we wish to populate with useful features for downstream classification. The word2vec objective attempts to predict, for a given center word in a sentence, what other words exist within a predefined window. The classification takes the center word representation as input, and uses it to classify a probability distribution over possible context words. This self-supervised learning task extracts useful information about words in context. To form a representation of a sentence, the mean word representation can be computed across the sequence.

While Word2Vec extracts useful information about individual words or tokens, it cannot effectively process the meaning of phrases or larger language structures. One example of this is word sense. For example, the word ”bank” has different meaning in the context of a river or a financial institution. We wish to have a model which learns *contextual* representations. One such model is BERT [10]. BERT operates by taking as input an entire sequence of tokens as input. Instead of predicting individual context words from a center word as in Word2Vec, BERT utilizes a masked language modelling objective whereby random tokens are masked as input. The BERT system is tasked with predicting these tokens at the output. In a sense, the model ”fills in the blanks” using its understanding of natural language. The output of the system is a sequence of word representations, the mean of which can be taken as a sentence representation. Alternatively, [7] fine-tune this mean representation on a supervised natural language inference task to produce a better sequence representation that is useful on downstream tasks such as sentiment classification or semantic similarity prediction.

[1] Github repo: `https://github.com/djd1283/UndiagnosedDiseaseClassifier`

While these BERT-based approaches are appropriate for everyday language, it may be useful to have clinical-domain knowledge for use in disease classification tasks. [1] train the BERT masked language modelling objective on a vast amount of clinical documents, and make clinical-BERT available to the public.

3 Method

Within this limited data domain, it is important to choose models which can learn from a limited number of examples. We outline possible models which may be robust to the limited data requirement. Any model of choice must map an input string of tokens (a Tweet, Reddit post, etc.) to an output label predicting whether this post is of an undiagnosed disease, or a regular Tweet (a negative example).

First, we could analyze the tweet using a bag-of-words approach, where word representations are looked up for each word in the post, and these are averaged to produce a representation as input to a linear classifier or support vector machine (SVM). This approach has minimal parameters, and may serve as a useful baseline. Second, we could include a larger pretrained model such as Sentence-BERT [7] or clinical-BERT [1]. These systems would analyzes the input text to produce an output representation of the sentence by calculating the mean across contextualized word representations. This representation would then be sent as input to a classifier.

However, in this work we manually annotate a small number of examples for training. The quantity of examples available for this task is less than that needed to train a linear classifier over hundreds of pretrained features from either word embeddings or contextual models. For this, we need a simpler approach that does not remove dependence on pretrained linguistic and world knowledge.

3.1 Linear Classification over Contextual Features

We design our system as a linear classification over seven selected input features. Each feature is weighted and the resulting sum is sent as input to a softmax-cross entropy loss objective for minimization to learn the proper weighting between features from the training data. The features we select are:

- SBERT key phrase similarity score
- GPT language model word-level score
- GPT key phrase score difference
- Log submission token length
- UDN keyword token indicator
- Doctor token indicator
- UDN Examples BLEU similarity score

We also include an additional clustering feature for the Twitter domain only, described in the unsupervised section 4.2.

3.1.1 SBERT key phrase similarity score

At a high-level, we want to know if the post contains an undiagnosed disease. A simple post which would satisfy this condition becomes our key phrase: "I have an undiagnosed disease". One assumption would be that any post which shares semantic content with this key phrase (it tries to convey a similar idea) is itself more likely to be a post about an undiagnosed disease.

We use Sentence-BERT [7] to produce semantic vector representations for each submission we intend to classify, and compare this representation to the computed representation of the key phrase with cosine similarity. Sentence-BERT uses the power of standard BERT [2] with large-scale pretraining to learn a broader measure of semantic meaning across sentences. Thus, the cosine similarity in its learned semantic space is a good indicator of similarity to the key phrase.

3.1.2 GPT language model word-level score

Next, we add another input feature computed from another pretrained model, the GPT language model [5] which computes the probability of each word occurring sequentially in a sequence. The objective function used to train GPT is word-level cross entropy over the input tokens, and we use this loss as a score of fluency/diversity among words in a submissions. Words which are more technical will have a lower probability and thus a higher GPT score because technical words are by definition less common than more colloquial ones. Similarly, all posts which are more diverse will have higher GPT scores, which may be a good indicator for posts in the medical domain with complex vocabularies.

3.1.3 GPT key phrase score difference

This feature is similar to the GPT feature about, but it instead measures the difference in score for a given input submission before and after the key phrase is appended to the end. If the post does not fit the key phrase, then it will be out of place and thus will have lower probability. This feature can be used similar to the SBERT feature to detect relevance to the key phrase.

3.1.4 Log submission token length

As a simplistic yet powerful feature, we include the log (normalized) length of each submission as an input feature.

3.1.5 UDN keyword token indicator

Next, from the UDN we have extracted a list of keywords which are unique to undiagnosed disease profiles. For each submission, it is checked to see whether any of these keywords appear in the post. If so, a binary label of 1 is assigned, otherwise 0.

3.1.6 Doctor token indicator

The submission is checked as to whether it contains the word "doctor" or "doctors". An undiagnosed disease post often talks about how they have visited multiple doctors with no positive results or diagnosis.

3.1.7 UDN Examples BLEU similarity score

Finally, we extract actual profile descriptions from the UDN network and analyze the BLEU score similarity between the descriptions and a given social media submission. BLEU score is a measure of n-gram word overlap between posts, averaged across all reference descriptions. If a social media post contains similar symptoms, it should have a higher BLEU score after being matched with one of the sequences.

3.2 Unsupervised Method

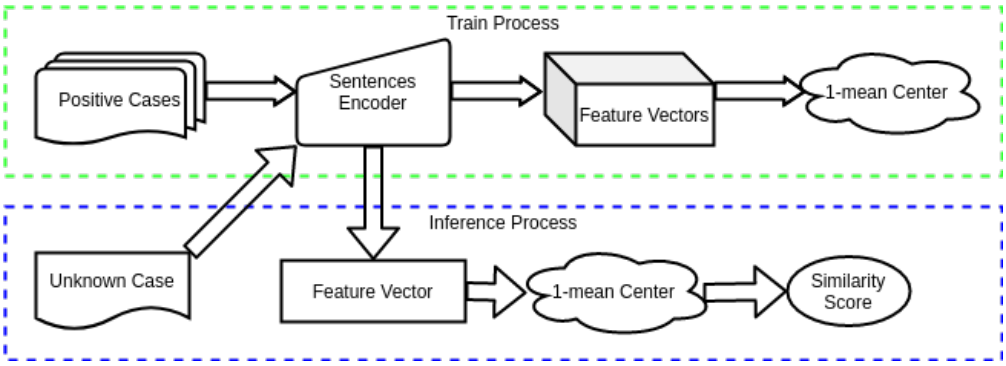

Figure 2: Work Flow of Unsupervised Method

Fig. 2 shows the work flow of our proposed unsupervised method. There are two stages in the work flow. The first stage contains the train process, in which we only use positive cases as the training dataset. After collecting the positive cases set, we will adopt a sentences encoder [7] to transfer the sentence items to feature vectors. With the help of the feature vector, we use k-mean algorithm [11] to calculate the cluster center of the positive feature vector set. In the k-mean algorithm, we set k as 1. The second stage is about inference. In the process of inference, the unlabeled case will be the input and it will be converted to a feature vector with the same sentences encoder mentioned in the train process. Then the feature vector will be used to calculate the similarity score [4] with the 1-mean center that generated in the training process.

4 Data Preparation

In this project, we identify multiple language datasets which may be useful for data in our task. These include tweets, Reddit posts, terms, and UDN patient descriptions. The number of potential data in each dataset is listed in Table 1.

Table 1: Various data sources available for training. For Twitter and Reddit, we report positively labeled submissions (negatives can be gather en-masse).

Dataset	tweets	reddits	terms	patient descriptions
Number	151	54	112	141

These datasets can provide the clues for searching the positive cases after tallying the frequency of the words. Table 9 and Table 10 shows the statistic result of the top 5 words in the

tweets and patient descriptions dataset mentioned in Table 1 respectively. We remove the common words such as of, the, a, an, I etc. so that these keywords can focus more on disease. Of these datasets, we use Reddit and Twitter for training and others for feature-building or analysis.

4.1 Reddit Data Extraction

Currently, we have created a script which calls the Reddit API to retrieve submissions from the subreddits r/health, r/undiagnosed, and r/diagnoseme. These serve as a starting point for collecting data. The script presents submissions to the user, which can label them as correctly identifying an undiagnosed disease or not. The script finally saves all accepted and rejected submissions to separate .tsv files, along with negative submissions (regular posts) extracted from non-medical subreddits to a different file.

While it may be beneficial to search Reddit dumps for relevant submissions, more recent dumps are too large to fit in our working environments. We may instead use dumps from 2014 or earlier which are small enough for download and extraction.

4.2 Twitter Data Extraction

Due to the shortage of Twitter data set, we develop a tool based on Twitter API [8] for Twitter data extraction. In the tool, we first create a script to extract more twitter text items. There are different opinions in the script. One opinion is to extract Twitter texts without keywords filter, which could help to collect negative cases with random texts. And other opinion is extract Twitter texts with keywords filter. In our case, we adopt the medical terms listed in Table. 1, which contains 93 terms. With the medical related filter, we could collect more positive Twitter texts. Another useful script in the tool is text cleaner. As the original text items from Twitter contains more than unicode texts and also the format haven't been unified, which will effect the generation of the feature vector. In the text cleaner script, we use several filter to figure out the none-unicode characters and the make each items unified. Another script is for manually selected the positive cases. This script will randomly select the text item one by one and show it on the screen and then give the reader three opinion, one is for positive cases and another is for negative cases. Due to the complexity of natural language, the items we read sometimes can be ambiguous, so in the process of manually selection, we use the third opinion to skip the ambiguous ones. We collect 100 positive cases with the medical terms, which are used in our training process.

5 Evaluation

Once the system is trained, we evaluate on held-out test sets manually annotated using the procedures above. We evaluate mean reciprocal rank, a measure of average reciprocal rank of the first positive example for that query. In our case, since there is only a single query "does this post contain an undiagnosed disease?", this measures the inverse rank of the highest-ranked undiagnosed disease post. If such a post has the highest rank, this metric is 1.0. For each data source, 1000 examples are selected as negatives and combined with all positively annotated examples. We also evaluate precision @ k for k = 5, 10, ,20, 30, 100, 500. This measures the precision of the classifier given the top-k highest ranked submissions over all examples. Put another way, this is a measure of how many undiagnosed disease posts made it to the top of the ranks produced by our classifier. These metrics are presented for the Reddit and Twitter domains in table 5.

To evaluate the importance of each input feature in the prediction task, we also perform an ablation study over each input feature. Specifically for each input feature, we provide only that

input feature as input and measure the objective loss value and accuracy after convergence of the classifier on that feature. To compute this accuracy measure, we selected an equal number of negative posts as positive posts such that 50% is random performance on this task. These values are reported on both the training and test sets, to show how certain features are only locally helpful on the training samples while others generalize to unseen submissions. These ablation tables are available for both Reddit and Twitter, in tables 5 and 6 respectively.

For the Twitter dataset, we also include the unsupervised cluster score described above in section 4.2.

In the appendix, we also include loss curves to visualize model behavior over the course of training, for completeness. These visualizations of training and accuracy curves for train and test sets (not used for hyperparameter tuning) are found in figures 3, 4, 5, and 6.

Dataset	R Rank	P@5	P@10	P@15	P@20	P@30	P@100	P@200	P@500
Twitter	**1.000**	**1.000**	0.900	**0.933**	**0.850**	**0.733**	**0.450**	**0.280**	**0.138**
Twitter + Cluster	**1.000**	**1.000**	**0.933**	0.900	0.767	0.490	0.290	0.140	0.071
Reddit	**1.000**	0.600	0.600	0.533	0.450	0.400	0.180	0.110	0.044

Table 2: This table contains our results when evaluating our trained undiagnosed disease classifier on the Reddit and Twitter domains. Positively annotated examples are combined with 1000 negative examples to calculate reciprocal rank (R Rank) and precision metrics @ k. Cluster feature did not change results

Input Feature	Train Cross Entropy	Train Accuracy	Test Cross Entropy	Test Accuracy
SBERT Sim	0.668	0.569	0.660	0.606
GPT Score	0.583	0.688	0.498	0.845
GPT Diff	0.692	0.587	0.688	0.627
Log Length	0.548	0.775	0.562	0.711
Keywords	**0.444**	0.794	0.711	0.549
Doctors	0.689	0.512	0.676	0.549
UDN Sim	0.567	**0.806**	0.538	0.880
Cluster	0.510	0.894	**0.497**	**0.944**

Table 3: Ablation study for different features as input to the model, as evaluated on the collected Twitter data.

Input Feature	Train Cross Entropy	Train Accuracy	Test Cross Entropy	Test Accuracy
SBERT Sim	0.659	0.810	0.670	0.729
GPT Score	0.620	0.707	0.619	0.750
GPT Diff	0.694	0.345	0.694	0.458
Log Length	0.634	0.707	0.603	0.792
Keywords	0.664	0.569	0.640	0.625
Doctors	**0.398**	**0.862**	**0.428**	**0.833**
UDN Sim	0.6748	0.741	0.672	0.729

Table 4: Ablation study for different features as input to the model, as evaluated on the collected Reddit data.

6 Results and Insights

Overall, the classifier performed well in classifying undiagnosed disease posts from other randomly sampled posts from Reddit and Twitter. The combination of logical, semantic and clinical features allowed the system to properly selected unseen diagnosed posts, with a reciprocal rank of 1.0 indicating that the top-ranked post was a positive example for both datasets. The Twitter dataset goes further, we all of the top-5 ranked posts being actual undiagnosed disease posts, and 90% of posts in the top 10. It seems that in the ranking task, the classifier performed better in identifying top posts in the Twitter domain than in the Reddit domain.

The unsupervised SBERT clustering score (Twitter + Cluster) had a large impact on the ranking task. This indicates that the semantic space of SBERT is rich enough to compare label undiagnosed disease posts with other posts we want to classify in euclidean space. The ablation study in the Twitter domain demonstrated that the cluster score had the most impact on test accuracy. This is intuitive as this feature captures the semantic relatedness between positive undiagnosed disease posts and other posts we wish to rank.

In addition to the cluster score, UDN similarity feature had the second-most impact on test accuracy. This is likely because keywords or clinical phrases were matched between Tweets and the Undiagnosed Disease Network profiles. However, it is possible this metric could be improved. BLEU measures average similarity to all UDN posts, but it is possible that the maximum similarity to any one UDN post is more indicative. If UDN posts are inherently different from one another, then a submission that is similar to one of them will receive low BLEU scores for the rest. This is an interesting direction for future features.

In the Reddit domain, the doctor indicator was the most powerful (yet most simple) feature. As the doctor indicator simply measures whether the post contains the word doctor, it would be well-suited for posts in which the patient complains that they have seen multiple doctors. Note that "doctor" was not a keyword used in the original data collection, but rather the keywords "undiagnosed" and "disease". However, in hindsight many posts contained the word doctor. When compared to randomized Reddit submissions, this could be a powerful discriminator as confirmed by the evaluation.

From the results above, it seems that technical indicators without underlying pretrained models were most helpful, althought features such as SBERT similarity were certainly helpful. While these evaluations do point to interesting findings as to what inputs are most supportive of predictive accuracy, it should be noted that these data sizes are small, and thus the level of uncertainty is high in these claims. In the Reddit domain, it was observed that the signs of certain feature weights in the classifier would flip if the training and validation splits were re-randomized, indicating that more data is needed. The effort of extensive data collection is left to future work, with this system as a highlight of future expectations. It is also possible that positive results on this task are easier when comparing undiagnosed disease posts to regular submissions, as this task simply requires discriminating between medical and non-medical. A more challenging task requires discriminating between undiagnosed disease posts within the medical domain.

Dataset	P@1	P@5	P@10	P@15	P@20	P@30	P@50	P@100
Twitter	1.000	1.000	0.700	0.733	0.700	0.767	0.780	0.800

Table 5: This table contains our results when evaluating our unsupervised method on the Twitter domains. P@K means positively annotated example percentage at the first ranked K cases.

7 Conclusion and Future Work

In conclusion, we create a classifier which is able to discriminate between posts which contain an undiagnosed disease and those which do not. Our system is evaluated in the Reddit and Twitter domains using hand-annotated data, and evaluated on a top-k ranking task with ablation studies to properly evaluate feature importance. Overall, our system is somewhat effective at finding undiagnosed disease posts among regular posts, but needs to be improved at scale.

In future work, more data is needed from both domains. With more data, we could switch to classification from bag-of-words and larger fine-tuning approaches using pretrained models. Otherwise, more features could be added and the UDN BLEU score feature could be improved to perform a max instead of a mean over individual BLEU features. Finally, more unsupervised approaches could be included for improved accuracy.

8 Contribution Charts

Task/Sub-task	Commentary on contribution
Twitter Data Crawler	Wrote code to crawl both positive and negative data from Twitter
Manually Labeling	Manually labeled positive and negative cases into two separate files
Introduction of trec eval	Wrote a detailed instruction of trec eval for use
Proposal and Progress Reports	Participated in writing these reports

Table 6: Contributions. Zijun He, Student ID: 01514642

Task/Sub-task	Commentary on contribution
Data Collection	Collect the initial data and statistic the items
Twitter Data Extraction	Develop the scripts for Twitter data extraction
Manually Annotation	Manually select 100 positive Twitter items
Negative Twitter Cases	Develop the script to get negative twitter cases
Similarity Scores	Create consine similarity scores for each twitter item
Unsupervised Method	Develop an unsupervised method based on k-mean
Proposal Report	Contributed a majority to the proposal report
Final Report	Contributed a considerable part to final report

Table 7: Contributions. Qilei Chen, Student ID: 01620563

Task/Sub-task	Commentary on contribution
Reddit collection	Collected Reddit submissions from both the API and Reddit dumps
Reddit labeling	Created labeling API and labeled Reddit posts as positive UDN examples
Model Creation	Created linear classifer and training script for Reddit and Twitter domains
Feature Creation	Created 7 features including SBERT score and keyword indicator
Trec Evaluation	Exported classifier predictions to file and performed ranking evaluation
Ablation	Performed ablation over features in both the Reddit and Twitter domains
Final Report	Contributed a majority to final report

Table 8: Contributions. David Donahue. Student ID: 01549608

References

[1] Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*, 2019.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[4] Hieu V Nguyen and Li Bai. Cosine similarity metric learning for face verification. In *Asian conference on computer vision*, pages 709–720. Springer, 2010.

[5] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.

[6] Rachel B Ramoni, John J Mulvihill, David R Adams, Patrick Allard, Euan A Ashley, Jonathan A Bernstein, William A Gahl, Rizwan Hamid, Joseph Loscalzo, Alexa T McCray, et al. The undiagnosed diseases network: accelerating discovery about health and disease. *The American Journal of Human Genetics*, 100(2):185–192, 2017.

[7] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[8] Ulf-Dietrich Reips and Pablo Garaizar. Mining twitter: A source for psychological wisdom of the crowds. *Behavior research methods*, 43(3):635, 2011.

[9] The UDN. The undiagnosed diaseases network. `https://undiagnosed.hms.harvard.edu/`. Accessed March 14, 2020.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[11] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. Constrained k-means clustering with background knowledge. In *Icml*, volume 1, pages 577–584, 2001.

9 Appendix

9.1 Terms for Keyword Feature

These are the UDN terms we search for during classification to form our keyword indicator feature:

weird condition, no diagnosis, no treatment, rare disease, rare condition, Mastocytosis, Chiari, cystic fibrosis, neuromuscular, abnormal MRI, drooping eyelids, involuntary movements, chorea, severe nearsightedness, myopia, double vision, diplopia, blurred vision, vision loss, visual impairment, small head size, large head size, microcephaly, painful redness, eye turning inward, alternating esotropia, involuntary movements, choreoathetosis, gastrointestinal dysmotility, bleeding disorder, bleeding disease, abnormal muscle contractions, abnormal dystonia, repetitive behaviors, stereotypy, muscle twitching, fasciculations, abnormal eye movements, saccadic smooth pursuit, difficulty in speaking, difficulty in walking, difficulty in growing, difficulty speaking, difficulty walking, difficulty growing, difficulty in swallowing, difficulty swallowing, dysphonia, weak bone, brittle bone, Osteogenesis imperfecta, intention tremor, pituitary hormones, panhypopituitarism, abnormal growths, abnormal bone growths, abnormal cartilage growths, blood clot in veins, blood clots in veins, thrombophlebitis, language delay, absent saliva flow, macrocephaly, pigmentation of hair, pigmentation of skin, pigmentation of the hair, pigmentation of the skin, hypopigmentation, ventriculomegaly, malrotation of bilateral hippocampi, decreased muscle tone, low muscle tone, hypotonia, underdeveloped optic nerve, optic nerve hypoplasia, mosaic Turner syndrome, dysmetria, right sided weakness, weak cartilage, laryngomalacia, autoimmune hemolytic anemia, antiphospholipid antibody positivity, myalgia, dysarthria, slurred speech, echolalia, absent speech, severe disease, undiagnosed, problem with coordination, problems with coordination, uncoordinated walking, ataxia, swelling on her brain, swelling on his brain, swelling on my brain, swelling on brain, delayed brain, brain disease, abnormal brain, brain anomalies, brain anomaly, brain abnormality, hypotonia, developmental delay, eye disease, abnormal eye, eye anomalies, eye anomaly, eye abnormality, microphthalmia, intention tremor

9.2 Analysis of UDN Keywords

Table 9: Top 5 keyworks in tweets.

Keyword	gene	delay	variants	seizures	variant
Frequency	17	17	15	13	12

Table 10: Top 5 keyworks in patient descriptions.

Keyword	developmental	delay	muscle	seizures	tone
Frequency	47	47	38	34	17

9.3 Visualizations/Plots

We plot losses and accuracy over time for both training and test sets.

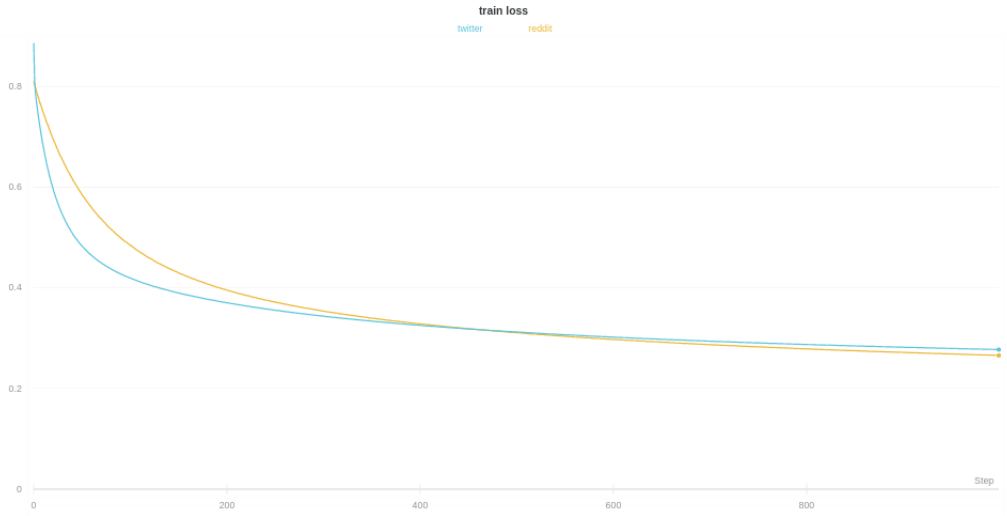

Figure 3: Train loss

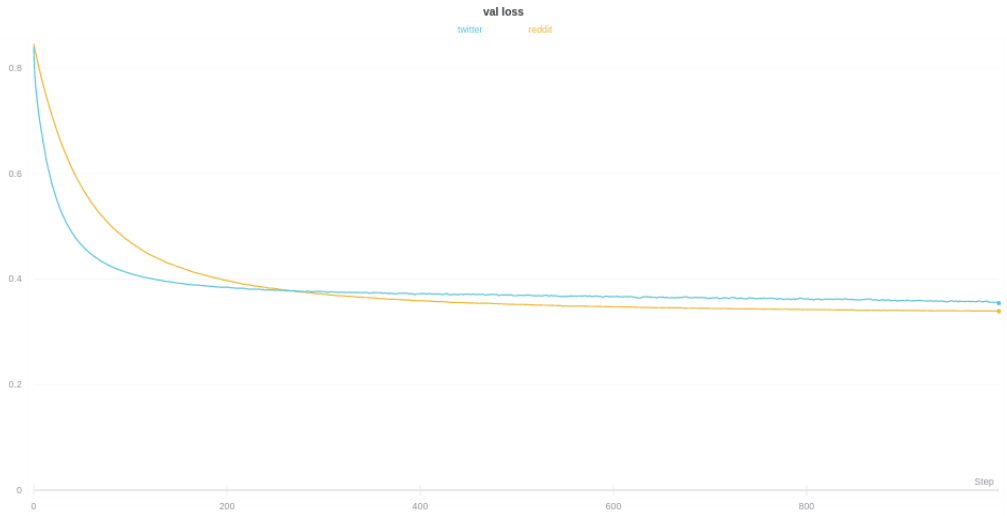

Figure 4: Test loss

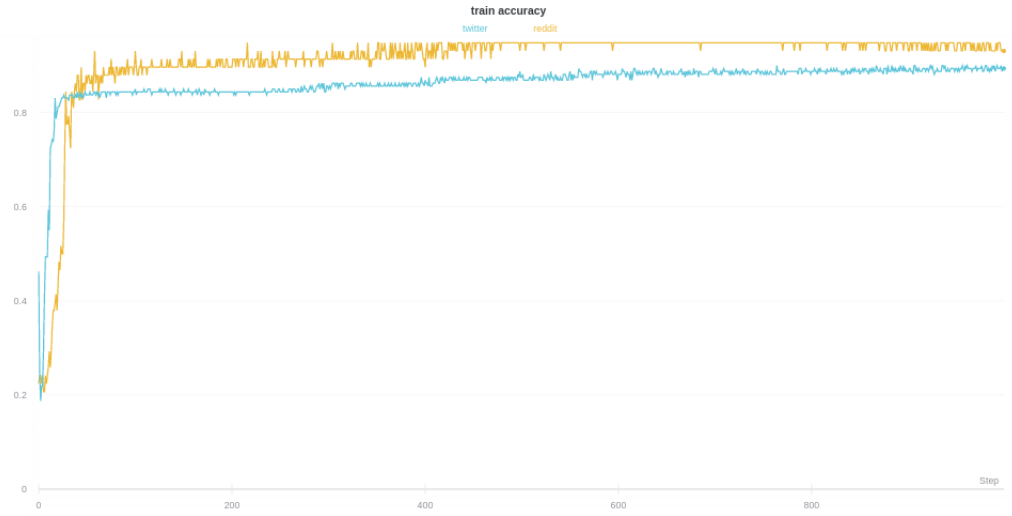

Figure 5: Train accuracy

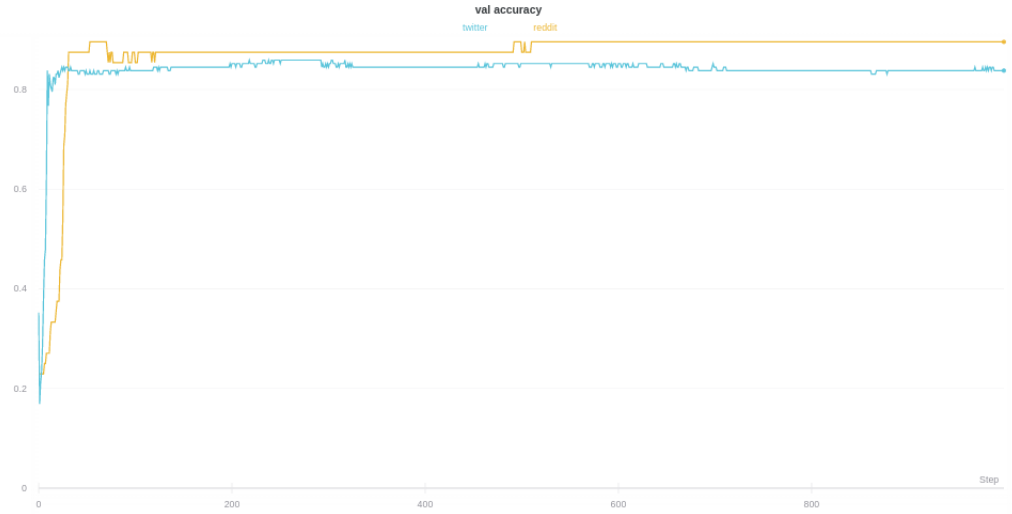

Figure 6: Test accuracy