The Reddit Self-Post Classification Task (RSPCT) : a highly multiclass dataset for text classification (PREPRINT)

Mike Swarbrick Jones
Evolution AI
mike@evolution.ai

Abstract

We introduce a publicly available dataset for text classification with 1013 classes and a large number of examples per class (1000), consisting of self-posts from Reddit. Posts are labelled by the 'subreddit' that they were posted in, and cover a wide range of topics, which we have classified and filtered to minimize overlap. We provide some benchmarks for text classification algorithms, and an analysis of how noisy the text/label pairs are from a human perspective.

1 Introduction

The goal of text categorization is to assign a class or multiple classes to natural language data based on its text content. Interest in this field has been growing steadily in recent decades — in machine learning, information retrieval and other related linguistic areas. It has found many practical applications, such as content filtering, recommendation engines and fraud detection. This growth has been met with an increasing number of publicly available text classification datasets.

The field of 'extreme classification', aims to classify or tag data when there are thousands or even millions of labels available, for example, the WikiLSHTC-325K dataset [4], or the AmazonCat-14K dataset [3] (number of labels being implied by the name). These datasets are extremely challenging for automated systems due to the extremely large label space, as well as data sparsity - most labels have very few exemplars in the data. For example, while there are hundreds of thousand of labels in the WikiLSHTC-325K dataset, there are only a few hundred which appear more than 100 times.

To contrast the situation to computer vision, we have the famous ILSRVC competition (ImageNet [5]) which was introduced in 2010, a dataset with 1000 classes, and roughly 1400 examples per class. It covers an extremely broad range of categories, from fine-grained dog breeds to unusual man-made objects. It helped to spark the deep learning revolution of the 2010s, with state of the art vision systems (convolutional neural network architectures) now outperforming humans on the competition's performance metric (see for example [1]). Our aim in this project was to create an interesting text classification dataset which covered a similarly broad range of topics, with a similar size and label distribution as ImageNet. Our data comes from the popular link-aggregating site `www.reddit.com`. On Reddit, users submit links to various 'subreddits' which are generally focused on one particular topic. The vast majority of these subreddits are created by users themselves, not by Reddit staff. There are two main types of posts on Reddit - external url links (such as images, videos, news articles etc.), and 'self-posts'. A self-post consists of a title, and (optionally) a block of markdown text. Generally the objective of self-posts is to generate discussion from other reddit users, in comment thread which accompanies every Reddit post. From ad-hoc analysis we realised that self-posts were a good candidate for a classification task rather than data from comments themselves, because they are more often recognizable without context ('on-topic'). An obvious classification task is to treat the subreddit as a label, and to try and predict this from the text content of the self-post.

It becomes clear that categorizing all posts into their subreddits is not generally feasible, due to substantial overlap in different subreddits' topics. Instead, we manually categorized a large set ($\geq$ 3000) of subreddits into a taxonomy which we then used to filter out the majority of them, resulting in posts from 1013 subreddits which we believe are suitably distinct in focus.

It must be emphasised that the self-post/label pairs have not been curated individually by humans (unlike datasets such as ImageNet). This means that classification is not feasible for all posts (this is problem for many datasets with large numbers of classes, see for example the del.icio.us tag dataset [8]). While this is an obvious drawback, there is some evidence that models trained on larger, noisier datasets can actually outperform those trained on manually labelled datasets (see [6]).

We will describe our methodology for obtaining the self-posts, creating the taxonomy, and minimizing label noise in section 2. In section 3 we will give some simple machine learning benchmarks, and analyze how often the posts were recognizably on-topic to a human, to get a rouch sense of the noise in our dataset.

2 The dataset

2.1 Data collection / filtering

We looked at all self-posts from 2016/06/01–2018/06/01. We filter out posts to try and ensure as many as possible have enough information to be classified by an English speaking person, and are normal and representative of the subreddit they are posted in. Posts that were filtered out included

- posts where the primary language detected was not English (as judged by the `langdetect` Python library).
- posts from subreddit moderators or Reddit admins, as well as posts from users suspected to be bots based on their username. We also filter out posts from users with more than 10 posts in a given subreddit.
- the body text (not including the title) was less than 256 or more than 4096 characters in length, after cleaning (see appendix A for more information).
- duplicate posts, based on the concatenation of the title and body text using the min-hash algorithm (we found most duplicated posts were ones that were periodically posted e.g. 'weekly discussion threads')

After this filtering, we took those subreddits which had at least 1000 self-posts remaining (over 3000). Once we have filtered by subreddit (see next section), we pseudo-randomly select 1000 posts from each remaining subreddit using a hashing function.

2.2 Subreddit classification and selection

It becomes clear from browsing the most popular subreddits that there is often a significant overlap in content. For example, there are no fewer than 27 subreddits which satisfy our criteria for the video game 'League of Legends' (popular characters in the game have their own subreddit). There is also a peculiar trend of popular subreddits having 'twin' subreddits which are almost identical in purpose, for example r/buildapc, r/buildapcforme, and r/buildmeapc. It is essentially impossible for a human to disambiguate self-posts from these. For this reason we tried to omit subreddits where

- the subreddit's topic is broad, or posts in the subreddit are often of a broad nature ('off-topic').
- the subreddit's topic is shared by another, larger subreddit.

The choice to only take one subreddit per topic was to make it easier to locate problematic subreddits, the choice to take the largest one was arbitrary.

A subreddit's 'topic' was a difficult concept to develop. We want to make it as granular as possible to make the problem interesting and challenging from a text classification perspective. On the other hand, we want to make topics broad enough so that subreddits that closely overlap can be identified

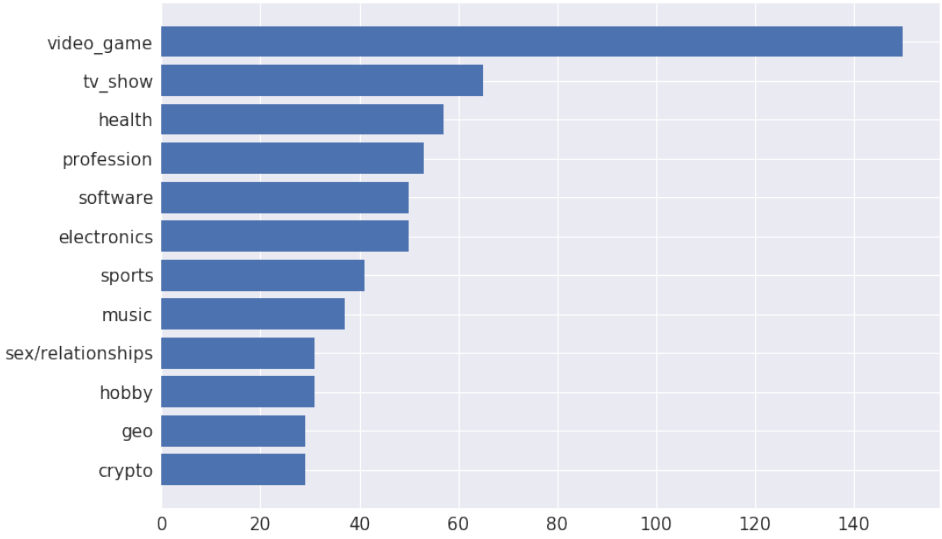

Figure 1: Breakdown of subreddits by category (top 12)

and the smaller ones omitted. This is necessarily a subjective process, however we tried to be as systematic as possible in how this was done.

We divide the subreddits into broad top-level categories, and then devised rules for each how to subdivide each category into smaller classes, in such a way that they can be disambiguated. These rules vary in their subjectivity, for full details of the categories and rules, see Appendix B. For some large categories we found evidence of significant label noise, so we made an attempt at additional filtering on posts from these subreddits (see Appendix A).

Our topic categorization provides us with a two-level hierarchy of topics. This hierarchy should be seen as byproduct of our initial goals, rather than the best taxonomy from a text classification perspective.

It should also be noted that the most popular subreddits tend to be broad in nature, so the majority of subreddits that will be familiar to most users are not included in this dataset.

We found that there were a very large number of subreddits dedicated to video games in our dataset (240). To lower data uniformity, we randomly downsampled the number of subreddits in our dataset concerning video games to 100.

In figure 1 we show a breakdown of some of the most common top-level categories in our dataset. It should be noted that the skew towards video games would be even more pronounced without the extra cleaning of posts from this category detailed in section C.

In figure 2 we show an unsupervised plot of posts grouped by subreddit, colored by top-level categories. This was produced by a simple bag-of-words approach, clustered by the t-SNE algorithm [7].

3 Benchmarks

3.1 Metrics

We use Precision-at-K (P@K) metrics to judge between models here – for each data-point in the test data, produce the K top labels $l_1, \ldots, l_K$. Score 1 if the ground truth label is in this set, else 0. Then precision-at-K is the average of this score over all data points. Though we have taken great pains to reduce the overlap between posts from different classes, some overlap is impossible to avoid, no

[1] to analyze this plot in more detail, see

Figure 2: t-SNE plot of subreddits, coloured by top-label category [1]

matter how coarse our categorization. For this reason, metrics relying on more than one output may be more suitable for this dataset.

3.2 Text classification algorithms

For each algorithm we concatenate the title and the text of each self post to get one string.

We take an 80/20 training/test split, stratified on the label.

3.2.1 Bag of word approaches

We try a couple of bag-of-words approaches:

Naive Bayes We vectorise each self-post using Tf-Idf weightings, on words and bigrams extracted from the text. We initially select the 100,000 most common features, and then reduce this to 30,000 based on the features which have the highest chi-squared score to our labels. We then use the Naive Bayes from `scikit-learn`, with smoothing parameter 0.1.

FastText [2], [2] - we do this in lieu of logistic regression, which is expensive to train, and to which this algorithm is very similar.

3.2.2 Sequential models

GRUs (3 stacked)

3.3 Results

Model	P@1	P@3	P@5
Naive Bayes	0.739	0.854	0.889
FastText	0.763	0.870	0.900
LSTM(GRU)	0.579	0.716	0.768

It is interesting that sequential models, which are generally more performant than bag-of-words approaches with large sample sizes such as with this dataset, underperform here. A few hypotheses for why this may be :

[2]implementation : (`https://github.com/facebookresearch/fastText`), trained for 100 epochs

- The information density of the Reddit posts is quite sparse - it is often the case that only one or two words in the self-post give away the class.

- Relatively few exemplars per class - while sequential models outperform bag-of-words given enough data, this tends to be when the number of classes is low (e.g. sentiment analysis). It could that sequential models are not good at building hierarchical representations of the text on this dataset.

3.4 Human judgement

Without manually classifying every self-post, it is impossible to avoid the fact that some posts will be impossible to classify (even using metrics such as precision@5), as there are no guarantees that the self-posts will be on topic, or give enough information to classify them. To get a rough idea of how feasible the task is, we took a sample of 1000 self-posts, and asked an annotator to judge whether or not they thought it was impossible to reliably guess the subreddit/topic from the title and body-text of the self-post. They thought that 42 were completely off-topic, or too broad to be guessable, leading us to guess that a limit on the error rate on this dataset should be in the 3.1% - 5.7% range (at least for metrics such as precision@5).

Acknowledgments

I'd like to thank my colleagues at Evolution AI for their encouragement and useful comments during the duration of this project.

References

[1] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *CoRR*, abs/1709.01507, 2017.

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

[3] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pages 165–172, New York, NY, USA, 2013. ACM.

[4] Ioannis Partalas, Aris Kosmopoulos, Nicolas Baskiotis, Thierry Artières, George Paliouras, Éric Gaussier, Ion Androutsopoulos, Massih-Reza Amini, and Patrick Gallinari. LSHTC: A benchmark for large-scale text classification. *CoRR*, abs/1503.08581, 2015.

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[6] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. *CoRR*, abs/1707.02968, 2017.

[7] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[8] Robert Wetzker, Carsten Zimmermann, and Christian Bauckhage. Analyzing social bookmarking systems: A del.icio.us cookbook. In *Mining Social Data (MSoDa) Workshop Proceedings*, pages 26–30. ECAI 2008, July 2008.

Appendices

A Text cleaning

One of our heuristics for whether or not a post was likely to be on-topic was the length of the body text of the self-post. After an ad-hoc analysis we decided to set a lower bound at 256 characters. However, there are posts that are longer than this only because they contain long bits of text that are not clean free-text. For this reason, prior to checking the length of the post we removed

- URLs, based on the following regex "`http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+`
- text inside html tags
- excess whitespace
- characters repeated more than twice consecutively

B Rules for categorization / exclusion

Here we will describe our first and second level topics (we will call them 'categories' and 'subcategories') we use to classify subreddits. We will give a brief description of the categories, and what criteria we have used to subcategorize them. It should be assumed that if the criteria does not make sense for a subreddit in the category, it is because the subreddit is too broad (e.g. movie subreddits that are not dedicated to a movie or movies from a creative universe, such as r/movies). These subreddits are omitted from our dataset.

There are still problems here. For example subreddits from from different categories are still easily confusable, e.g. self-posts from TV shows from the DC Comic Universe are easily confused with video games, movies or comics from the same universe. For this reason, we also tried to label the subcategories in such a way that they can be identified across categories, and the smaller duplicates removed.

video games subreddits dedicated to talking about individual video games, or video game series.

We subcategorize these based on individually released games, or series of games, or the creative universe these games take place in, if this would lead to conflicts.

electronics subreddits dedicated to talking about electronic devices.

We subcategorize these subreddits into what type of electronic device is being discussed (or inferred, such as vaping juice subreddits being used with electronic cigarettes).

software/os subreddits talking about software or operating systems.

Subcategories are the purpose of the software. We also split operating systems into the following 4 subclasses - android, unix/unix-like, windows, apple.

hardware/tools subreddits talking about non-electronic hardware or tools.

Subcategories are the type / purpose of the piece of hardware or tool being discussed. The next level of categorisation occurs if a specific device or range of devices from a company is being discussed, with the tag being the name of the company.

company subreddits discussing companies or websites (not covered by electronics / video games etc.)

Subcategories are the main business activity of the company.

tv_show subreddits talking about TV shows not elsewhere classified.

Subcategories are the universe in which tv show is set (just the name of the TV show if this is not shared by another category)

health subreddits discussing aspects of human health.

We subcategorize based on either : the physical or mental disorder being discussed, or the treatment being administered, or part of the body affected. We take care to make sure that the subreddits are not obvious symptoms of another disorder (e.g. depression being a symptom of many mental disorders). The biggest subcategory was subreddits dedicated to various diets, which we grouped into one - diet.

music subreddits dedicated to music. 1st level - instrument, musical act or musical genre, which we disambiguate from the genre's Wikipedia page (e.g. we classify 'emo' music as 'rock'.

profession subreddits dedicated to professions.

We subcategorize based on profession. Posts in these subreddits tend to be asking for expert advice by laymen, or career advice from other or future professionals. We removed subreddits referring to mechanic advice, as we found this heavily overlapped with the 'autos' category

sports subreddits dedicated to sports and fitness. 1st level - sport being played. We note that there are many subreddits dedicated to sports teams (especially American football and baseball). We decided to group these under the single category. We also remove fantasy-game subreddits (e.g. fantasy football), as we found these were often not talking about the sport they refer to themselves.

writing/stories A fairly broad category dedicated to anecdotes and other writing. The top level is the aspect of writing or the type of writing being discussed. subreddits dedicated to telling stories or anecdotes we classify based on the general theme of these stories (e.g. 'confession').

hobby Pastimes not covered by profession, sports or game-based subreddits. We choose the subclasses based on the pastime.

crypto Discussing cryptocurrencies. 1st level - the cryptocurrency or aspect of cryptocurrencies being discussed.

arts_crafts subreddits dedicated to the arts (performance or visual). Subcategories are chosen based on the artistic technique, or type of performance art being discussed.

politics/viewpoint Top level is a distinct political viewpoint (must have its own Wikipedia page). We exclude subreddits devoted to individual politicians or other political figures here, as we find they overlapped with broader subreddits too often.

sex/relationships subreddits dedicated to relationships and sexual preferences. The subclasses are chosen based on the sexual preference or type of relationship advice being discussed. A large number of these subreddits are r4r, which we remove.

anime/manga Discussing anime/manga comics and animations.

Subcategories are the universe in which the anime/manga take place.

drugs Discussing drugs. Subcategories are the drug, or type of drug being discussed (e.g. opiates), as well as ways to obtain drugs.

autos subreddits discussing brands of automobiles.

Subcategories are the manufacturer.

social_group This is a broad category encompassing groups of people who share a common trait, not elsewhere covered.

Subcategories are our judgment about this common trait.

programming discussing programming.

Subcategories are the programming language being discussed

podcast/video_blog discussing podcast channels, YouTube channels, etc.

It should be noted that we completely omitted this category from our dataset, as we found that users very often use these subreddits as forums to connect with other people with the same interests.

animals discussing animals.

Subcategories are the species, or broad class of animals.

stem discussing scientific, engineering and mathematical pursuits. We also put

Subcategories are the

education Discussing education.

Subcategories are the type of education (the level of education if this is in school or college). It should be noted that there were many subreddits dedicated to individual colleges/universities, and we group these all into one subcategory.

appearance subreddits dedicated to fashion, makeup and grooming.

Subcategories are the type of fashion, or aspect of grooming.

food/drink subreddits dedicated to food, drink and cooking.

Subcategories are the type of cooking, food or drink.

religion/supernatural subreddits dedicated to supernatural beliefs.

Subcategories are the religion or type of supernatural phenomenon being discussed. We remove subreddits where the topic is *not* having a particular supernatural belief (e.g. ex-Muslims), as we saw that these subreddits share a lot of overlap in content.

advice/question subreddits dedicated to asking for questions and advice, not elsewhere covered by the other subreddits.

We subcategorize based on the type of question being asked.

card_game subreddits dedicated to card games (not card-game based video games).

Subcategories are the universe in which the card game exists

parenting subreddits discussing aspects of parenting.

Subcategories are the aspect of parenting being discussed. Broadly we break this down into which stage in the parenting process of this aspect (from conception to childhood)

books subreddits dedicated to books and (western) comic books

First level is the universe the book takes place in (if fiction), otherwise the book itself.

movies Discussing movies.

First level is the universe the movie takes place in.

meta Discussing or parodying Reddit itself.

Subcategories are the type of discussion (we pull out parody into its own category).

board_game Discussing table-top games

First level is the universe the 'universe' the game lives in e.g. lives in, if it has a narrative element, else just the game e.g. go (baduk).

finance/money Aspects of finance and money.

We group together all subreddits involving some type of investing, as we found these had substantial overlap.

rpg Discussing role playing games

We subcategorize based on the 'universe' that the game takes place in (such as Dungeons and Dragons).

travel Discussing aspects of travel.

We subcategorize based on the aspect, or type of travel being discussed.

C Category based post filtering

We noticed some patterns in off-topic posts in some of the larger top-level categories. To reduce label noise on these, we filter posts based on simple string matching (after lower-casing).

Clearly, this adds bias and a level of syntheticity to the dataset. However we decided this was a reasonable price to pay to reduce noise, which we measured empirically. It also had the unintended (though perhaps welcome) effect of reducing the skew of the dataset towards videogame-based subreddits, as these took up approximately 35% of the subreddits before the filtering, and about 15% afterwards.

video_game: we noticed that many posts were asking for technical help (troubleshooting) in getting games to run on particular hardware, rather than discussing the game itself. It was often impossible to distinguish the video game being discussed from this. We also try to remove posts which are complaining in a general sense about poor user experiences playing online. We filter posts containing the following strings (after lower-casing).

```
bug, connection, patch, resolution, screen, glitch, launcher,
framerate, frames, fps, update, crash, lobby, matchmak, latency,
black screen, issue, steam, desync, gtx, cheat, file, download,
upload
```

music: many posts were asking about particular shows of particular bands. These were often trying to re-sell tickets, without explicitly mentioning the name of the band. We filter posts containing the following strings.

```
tour, ticket, concert, show, venue
```