Extreme Multi-Label Text Classification

Astha Mishra, Manav Jain, Megha Bagri, Siddhant Gupta, Sichuo Xie

asthamis@usc.edu, manavpra@usc.edu, mbagri@usc.edu, sgupta86@usc.edu, shicuoxi@usc.edu

1 Motivation

The problem of extreme multi-label text classification (XMTC) is an important yet very challenging problem in NLP, specifically because of its ubiquitous applications such as Information Retrieval, User Profiling, Online Advertising, and Web Searches. XMTC is a multi-label text classification problem in which the labels can scale up to millions. This makes the problem intractable.

The application of neural language models to solve this problem is relatively new. X-Transformer [Chang et al., 2019], LightXML [Jiang et al., 2021], RankAE [Wang *et al.*, 2019], and [Mittal et al., 2021a] are some of the initial works in this area. The model proposed by [Wang et al., 2022], which utilizes label semantics through a guided network (GUDN), has achieved state-of-the-art accuracy.

[Wang *et al.*, 2022] tested GUDN on Eurlex-4K, AmazonCat-13K, Wiki10-31K, and Wiki-500K. The problem with these datasets is that they are very verbose and domain-specific. GUDN achieves the best performance on the Eurolex-4k dataset, which has an overlap between text and labels. Since GUDN explores the latent space between the label and text, a strong relation between them provides an added benefit to the model and thus doesn't evaluate it completely. Based on these observations and the problems, we have decided to construct a dataset that will overcome these issues.

1.1 Novel Contributions

Twitter seems like a potential data source for this problem statement. We plan to create a dataset of tweets, where the hashtags work as labels.

The tweets are from an enormous variety of domains, which will help train the model for generic scenarios. The number of hashtags (labels) associated with the tweets accounts for a huge label space, which creates a case for XMTC. The length of a tweet is limited to 280 words and the tags are weakly linked to the tweet text, which is often the case for real-world data.

2 Design

2.1 Materials

We plan to use the Twitter API for data scraping. We aim to scrape at least 18k tweets and scale the volume of data depending upon how much time it takes in the post-processing of the data.

2.2 Method

Domains: We plan to collect tweets related to multiple domains. We do this by selecting some popular labels related to the domains and scraping the tweets that come up.

Filtering: We will filter out the tweets containing media such as images and videos as the proposed model cannot process such data. All the tweets in languages other than English will also be filtered. Hence, we will only consider tweets containing pure text and written in English.

Modes: We plan to split the data into easy and hard modules. The easy module will be a smaller dataset that will be more idealistic. The text will be often strongly linked to the tweets and the tweets will be relevant to the domain. This mode is expected to have fewer inconsistencies. The harder version will be closer to real-life data and will not be curated.

Test: We partition our dataset into multiple domains, which enables the user to train on specific domains and test on unseen domains. This will test if the model learned is generic enough to handle unseen data.

2.3 Baselines and Evaluation Protocols

2.3.1 Baselines

We plan to evaluate our dataset on the most recent neural models - X-Transformer, LightXML, GUDN.

2.3.2 Evaluation Protocols

In XMTC, although the label space is huge, there are only a few relevant labels for each text. That is why it is crucial to capture this shortlist of the most relevant labels per text. For this reason, rank-based evaluation metrics are widely used in XMTC tasks. These metrics emphasize the top k most relevant labels of a text. We are using P@k which represents the precision of the top k labels.

We will use P@k to test the above mentioned models on our dataset and compare their performance with Eurlex-4K, AmazonCat-13K, Wiki10-31K, and Wiki-500K.

3 Timeframe and Division of Work

3.1 Time frame

Mar 9	Project proposal
Mar 16 - 31	Collection and cleaning of Twitter data
Mar 31	Demo
Apr 7 - 20	Implementation of neural models
Apr 20	Poster presentation
Apr 29	Final submission

3.2 Division of work

Astha Mishra	Implementing the LightXML model. Evaluate and document the results.
Manav Jain	Implementing the GUDN model. Evaluate and document the results.
Megha Bagri	Implementing the X-Transformer model. Evaluate and document the results.
Siddhant Gupta	Collection and cleaning of Twitter data.
Sichuo Xie	Collection and cleaning of Twitter data

Analysis

The model described is based on deep neural networks that require heavy computation and are difficult to train. We will need specialized hardware for this task. The model also mentions some minute optimizations that are not delineated. For example, the label clustering for a large number of labels is not described in detail which can make its implementation difficult. Furthermore, feasibility issues might arise due to a lack of GPUs and an error-prone data set. The real-world data is expected to have inconsistencies and errors. For example, a hashtag can be written in various formats, such as camel case, or lowercase. Tweets are often written in shorthand format that is not a standard and increases the chances of errors. In the worst case, the lack of hardware can be overcome by using cloud servers. Moreover, the data collection task poses a great risk. If the collected data has errors, manual cleaning will be required. This can limit the size of the data set and change our timelines drastically.

References

[Chang et al., 2022] Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit Dhillon. GUDN: A novel guide network for extreme multi-label text classification, arXiv:2201.11582v1, 2022.

[Chang et al., 2019] Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit Dhillon. Taming pretrained transformers for extreme multi-label text classification, arxiv:1905.02331, 2019.

[Mittal et al., 2021a] Anshul Mittal, Kunal Dahiya, Sheshansh Agrawal, Deepak Saini, Sumeet Agarwal, Purushottam Kar, and Manik Varma. Decaf: Deep extreme classification with label features, arxiv:2108.00368. arXiv e-prints, 2021.

[Yeh et al., 2017] Chih-Kuan Yeh, Wei-Chieh Wu, Wei-Jen Ko, and Yu-Chiang Frank Wang. Learning deep latent spaces for multi-label classification, arxiv:1707.00418, 2017.

[Jiang et al., 2021] Ting Jiang, Deqing Wang, Leilei Sun, Huayi Yang, Zhengyang Zhao, and Fuzhen Zhuang. Lightxml: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification, arxiv:2101.03305, 2021.