Text Classification Lit Review

Thomas Boyden thomasb3@stanford.edu
Robert DiDonato rdido@stanford.edu
Miles Zoltak mzoltak@stanford.edu

1 General problem/task definition

The papers reviewed here investigate the field of text classification. They cover many different areas of the field, from surveys on different algorithms and data structures to assist with feature extraction, data augmentation, and overall performance.

These are valuable contributions to the field because text classification is an important task in natural language processing. With massive amounts of textual data already extant and an ever-increasing pace of production of textual data it is as important as ever to have effective text classification techniques.

Broadly, text classification is the task of categorizing text into one of several categories, although it can sometimes involve assigning some subset of categories to a single document as seen in multi-class text classification. Types of classifications include authorship determination, topic assignments, sentiment analysis, and more.

In this literature review, 9 different articles spanning a wide variety of topics, applications, and methods within text classification are summarized and compared. At the end, the Future Work section expounds on some of the open questions posed by these papers with the hope of narrowing our focus toward a final project.

2 Article Summaries

2.1 Text Categorization Techniques

Big data refers to a large volume of highly complex data that can not be easily analyzed. The internet has introduced all sorts of distinct data types, meaning that in order to make sense of big data we categorize them into structured, unstructured, and semi-structured data types. [4]

Machine Learning algorithms are designed to learn from data in order to improve performance over time, and it does so in a variety of ways. One of these methods is supervised learning, which revolved around using labeled data mapped to its correct output, unsupervised learning which uses unstructured, unlabeled data, and reinforcement learning, which involves learning through interaction with the external world to learn and improve.

Data that comes in a declarative form, such as comments, tweets, write-ups, etc. is considered unstructured data, that is analyzed in natural language processing to obtain insights using sentiment analysis, machine translation, text summarization, opinion analysis, information extraction, and information retrieval.

Text categorization is a natural language processing technique that involves classifying documents of text into defined categories using supervised machine learning. Text mining is another technique to summarize, cluster, or categorize text using rules about textual patterns, and different approaches have been used to classify text in the Tamil and Marathi languages.

Algorithms such as Max-Ent, Conditional Random Fields, and Support Vector Machine have been compared for use in the case of the Tamil language, which browsing history analysis and user interests have been used for the Marathi language, but stemming has shown significant performance improvement when it comes to classifying text. The literature discusses a comparative study on text document classification technique in the various Indian languages.

These techniques include feature selection, clustering, key combination, POS tagging, and Neural Networks, using Telugu, Hindi, Marathi, and Tamil documents in the dataset to be used for classification. Dimensionality reduction techniques are evaluated for their use in real-world data, and it is shown that applying these techniques can improve the performance of the text categorization by extracting certain relevant features from the large datasets and reducing the computational and storage costs of the analysis.

2.2 Text Classification: From Shallow to Deep Learning

This paper provides an overview of the existing models that are used for text classification tasks, ranging from traditional text models to deep learning techniques. [5] Traditional models typically focus on the improvement of feature extraction schemes and classification design to improve the performance of text classification. In contrast, deep learning models use learning methods, additional data and training, and unique model structures to attempt to outperform the traditional methods.

The literature also explores datasets for single and multi-label tasks in addition to evaluation metrics. The paper additionally provides quantitative results when testing on datasets such as MR, SST-2. IMDB, Yelp.P, Yelp.F, Amazon.F, 20NG, AG, DBpedia, and SNLI.

The pre-trained models, including BERT, RoBERTa, and XLNET typically yielded more impressive results on most of the standardized datasets mentioned above with the exception of MR and 20NG, which have not previously been experimented with BERT models. The pre-trained models seem to improve the accuracy of natural language processing, due to their better learning due to the text features from training on unlabeled datasets.

It is worth noting that the RNN-Capsule was able to achieve the best accuracy of 83.8% on the MR dataset, which indicates its capability in the task of sentiment analysis. The RNN-Capsule does not rely on linguistic knowledge but rather uses capsules for each category of sentiment to achieve these exceptional results.

2.3 A Survey of Multi-label Text Classification Based on Deep Learning

This paper explores the task in natural language processing that is known as text classification, which involves the process of taking text input and categorizing it into groups that are organized as a method to analyze this text. [2] Within this technique of natural language processing, multi-label text classification is a method within text classification that has gained popularity. This technique involves assigning multiple labels to each data sample that is input to be analyzed.

The performance and accuracy of multi-label text classification has been vastly improved due to pre-trained models such as the BERT model. However, multi-label text classification still faces quite a few challenges. Firstly, the lack of datasets is a current challenge in multi-label text classification. This lack of high-quality datasets has stunted the improvement of text classification models. Another challenge is that multi-label text classification requires an increased amount of resources in comparison to a single-label text classification.

According to the paper, creating datasets of high quality is important for seeing improvement in the quality of models and to see progress in this space. Another challenge that multi-label text classification faces is the dynamic division of text-related labels. When labels change, under the current system, the models that rely on supervised learning require complete retraining. The relabeling of the dataset and the training of the model once again cause there to be high costs to the user, making it highly inefficient to do so.

The field would be greatly benefited by there being an efficient method at a low cost to the user for the model to be adapted to label changes.

2.4 Effective Feature Selection Approaches for Text Sentiment Classification Process

This piece focuses on the relationship between feature selection and the classification of sentiment in text. [6] The piece starts off by explaining the complexity of feature engineering due to the often large dimensionality of the data, making it increasingly harder to train models. The paper presents the idea of dimensionality reduction as a solution to this issue, detailing its two components: feature extraction, and feature selection.

When it comes to reducing the total amount of features, engineers want to make sure that only non-relevant features are disregarded, and that core features relating to the overall sentiment of a text are preserved. Such feature selection methods include Information Gain (IG), Mutual Information (MI), Chi square (CHI), Gain Ratio (GR), Document Frequency (DF) etc. (Tan S. and Zhang J., 2008; Pang B. and Lee L., 2008). The paper mentions how it's been discovered that EWGA and combined genetic algorithms have increased the precision aspects of sentiment classification. Furthermore, it's mentioned how there is an increase in unlabeled texts when compared to labeled ones due to small samples of text having large dimensionality aspects. The use of local methods for the feature selection process is also discussed, utilizing

the Odds Ratio, measuring the feature that falls into the positive category.

Later in the comparative study section, Word2Vec and Glove are also mentioned as strong solutions when it comes to extracting sentiment from a set of text, with the capability of use on videos and images as well. Overall, the general idea expressed by this paper is that feature selection techniques are vital when it comes to managing the millions of features in places such as web pages and emails when it comes to determining the overall sentiment of these datasets. This concept can be especially valuable in identifying sentiments from small datasets by OOVs (Out-of-Vocabulary Words). However, the one caveat with this system is that feature selection tends to have problems when used on small datasets with high dimensionality. For this issue, the paper recommends the use of MCDM methods.

2.5 Hypernym Techniques for Text Classification

This piece focuses on the issue of text classification through the means of AI utilizing feature expansion, particularly when implemented on unstructured datasets from social media sites such as Facebook, Linkedin, and Twitter. [7] The paper uses the term "text mining" to describe this process of transforming unstructured text data into accurate and relevant data. The article compares various different text classification techniques placing them into two categories: without feature expansion or with Hypernym-Hyponym based feature expansions. Automating the processing of large unstructured datasets is increasingly valuable to large social media platforms as their user base continues to grow, as such processing can enable these companies to make data-driven decisions based on the insights detected from this text mining. Oftentimes, large unstructured datasets from social media companies will contain issues such as spelling errors, making it increasingly complex for a less robust model to process and classify such datasets accurately.

The paper then explains the importance of feature expansion, specifically when dealing with unstructured textual datasets, due to the necessity for the text to be converted into numbers before being fed to the model. This leads to the introduction of the HyperRank algorithm, an algorithm that relies on the linguistic relation known as a Hypernym. This relation describes an overarching term that often denotes a subset of terms (eg. color is a hypernym of red, yellow, blue, etc.). The inverse of a hypernym is a hyponym (eg. red is a hyponym of color). Essentially, this algorithm works by taking input words and assuming them to be candidate hyponyms in order to find the overall hypernyms. In order to find the correct hypernyms, the algorithm utilizes the distance between a current (or groups) of candidate words with the nearest words in the feature vector induced by the word embedding. After the algorithm executes its three main components, evaluation can take place. Benchmark SemEval 2018 was used to determine that the algorithm was best amongst all unsupervised methods for this problem while also outperforming 50

This hypernym-hyponym based algorithm is then compared to other similar approaches using feature expansion such as LDA and FP Growth Algorithm, Pattern and Semantic Similarity based method, and Naïve Bayes Algorithm with Feature Expansion. Ultimately, the paper arrives at a couple conclusions. First being that feature expansion is highly necessary when it comes to the processing of unstructured textual data, largely due to the fact that features extracted from count-based vectorization methods omit semantics. Additionally, it's concluded that the primary reason for which the different algorithms mentioned above performed differently was due to the difference in data gathered.

2.6 Graph Neural Networks for Text Classification

The article initiates by explaining the transition within text classification approaches from the more traditional forms of machine learning models such as SVM along with the use of N-gram and TF-IDF to the more robust system of neural networks citing CNNs, RNNs, and LLMs. [8] However, it's stated that these newer approaches are still unable to handle the relationships between words and documents along with the inability to explore the contextual-aware word relations efficiently, hence where graph neural networks (GNNs) come into play. In order to utilize a GNN, a corpus level graph(s) or document level graph(s) must be constructed for a given dataset.

The piece then dives into great detail surrounding the mathematical components representing a

GNN such as word and document nodes, extra topic nodes, and single layer topic nodes. When it comes to the actual performance of these GNNs, the most common datasets they get tested on are 20NG, R8, R52, Ohsumed, and MR. Based on the corresponding performance table, multiple conclusions can be drawn regarding such models. First off, Models using external resources tend to achieve better performance than those that do not (this effect can be seen the most with BERT and RoBERTa). Furthermore, when used in identical settings, document-level GNNs tend to be outperformed by corpus-level GNNs. Finally, this advantage of Corpus-level GNN models over Document-level GNN models only relates to topic classification datasets (omitting sentiment analysis datasets). This is primarily due to sentiment analysis involvement in analyzing the order of words within a text, something that doesn't take place with Corpus-level GNNs.

Although GNNs possess much deeper relationships between words compared to standard ML techniques, there are still many complications with this approach including graph construction complexity, and applying GNNs without a pre-training format. Overall, GNNs seem to be quite promising with regard to their ability to get a "deeper" look at a field of text; however, there are still many intricacies that need to be addressed going forward in order to achieve optimal results.

2.7 Text Classification Using Embeddings

In this article, the authors note that the very common Bag of Words (BoW) model for text classification, which regards documents as an unordered set of words which may be a subset of a larger vocabulary, has shortcomings that can be addressed by using embeddings. [3] Specifically, where word order conveys information about context and word sense, embeddings specifically help overcome this hurdle by mapping words (usually) into sparse or dense vector representations. These vectors have the added benefit of being more effective for manipulation by neural networks, one of the most powerful methods for text classification.

The authors explore various types of embeddings and their strengths. Although the paper mostly focuses on word embeddings (which are by far the most common type of embeddings in NLP), the authors note the existence of other embeddings like knowledge embeddings. Knowledge embeddings codify the information inside a knowledge graph, or KG. KGs are semantic networks that encode relations between objects, situations, events, or concepts. As such, they provide even more complete contextual and semantic information than word vectors do. Later in the article the authors note that knowledge embeddings have the potential to go even beyond the impact that word embeddings have had in the text classification field. One reason that they have thus far come upshort is due to the fact that word and knowledge embeddings are trained independently and as a result end up in different vector spaces which hinders joint use. This is a potential avenue for further research.

In general, the article covers several different word embeddings, such as Word2Vec, GloVe, and BERT. With GloVe, the authors noted the high performance of capsule networks over traditional neural networks. Subsequently, the authors credited BERT with the ability to score over 90% regardless of model, so it may not be extremely important to lean on the benefits of the capsule networks. The article ended with a reflection on the fact that effective multi-label classification, a subset of text classification tasks in which documents are given several labels as compared to being placed into only one class, needs further work because the need for contemporary big and complex data structures is only increasing.

2.8 Data Augmentation for Text Classification

This article gives a very comprehensive overview of the uses of data augmentation for text classification. [1] Data augmentation (DA) is a valuable and important area of research because DA allows can serve as a tool for regularization, minimizing labeling effort, lowering the burden of collecting real-world data (especially in privacy-crucial contexts like healthcare), class balancing, and increasing robustness to adversarial challenges. One issue with data augmentation, especially with the rise of large pretrained language models, is that some models may already be invariant to the types of transformations seen in data augmentation, and so DA offers no benefit to the model. The authors cite this conclusion from Longpre et al, which suggests that for large pretrained models, augmentation must "create new linguistic patterns not seen before".

Data augmentation is seen commonly in computer vision and auditory processing contexts. Transformations here are intuitive: rotations, color changes, reflections for computer vision; variations

in pitch and speed for audio. In text classification the options for effective data augmentation are less clear. The authors dive deeply into the possible options and hierarchical levels at which data augmentation can be applied. Among these tiers is at the character level (slight perturbations to individual characters), at the word level (perhaps synonym or embeddings replacement), phrase and sentence level (interpolations and grammatical alterations come into play), or even at the document level (generative methods and "round-trip translation"). These are all examples of augmentations in the data space. Once words text is brought into the feature space, often via embeddings, noise introduction and interpolation can be employed here. Combinations of transformations in either or both of these spaces are also employable and have been successfully applied.

Overall, the article explains at a high level the benefits of data augmentation for text classification, how data augmentation might be performed and perhaps improved, and briefly attends to the potential drawbacks which include potential bias magnification as well as general time and cost consumption.

2.9 Topic Models in Text Classification

This article gives an overview of various types of topic models in the field of text classification. Topic models, and specifically LDA (Latent Dirichlet Allocation) are very useful in the modern field of NLP. [9] A topic model regards a set of documents D and a set of terms W from D. A set T of latent topics is learned by statistical inference on W. In short, each document belongs to some set of topics where each topic is a probability distribution over words.

Various types of topic models include Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA, the darling of topic modeling), Correlated Topic Models (CTM), and Dynamic Topic Models (DTM). Each method improves upon some shortcoming of the one before it. LDA benefits over LSA because it can capture the "exchangeability of both words and documents", but LDA is unable to model correlations between topics, where CTM accels. Sometimes, however, the assumption of exchangeable documents is not appropriate because documents may reflect evolving content - like in a news article; DTM offers a solution. The article provides a very good base for any investigative efforts into the strengths and limitations of any given prominent topic modeling method.

3 Compare and contrast

Several of these papers deal with rather disjoint topics, so to say that they agree or disagree with each other is not an entirely useful question. However, there were general trends that came up across several different articles.

The first is that the Bag of Words alone model is insufficient for effectively employing text classification methods because it fails to capture semantic relations between words and within sentences. Fortunately, there are many feature extraction methods that help bring semantic relations back into the text - specifically word embeddings. Across several papers, BERT and RoBERTa were acknowledged to be among the most powerful of these tools, although GloVe is still a powerful tool.

Similarly, the articles also all put forth deep neural networks as extremely adaptable and effective models for text classification (as well as many other tasks within and beyond NLP). Another architecture that came up in several articles in a positive light was capsule networks, which are commonly used in computer vision as a means of learning the hierarchical relationships that are present in object detection and recognition. However, language is also hierarchical and capsule networks appear to be effective tools in NLP and text classification as well.

More work is needed to explore if capsule networks are particularly beneficial for certain tasks or types of text classification. This can be seen in one of the few noticeable disagreements between papers, which stated that BERT can be so effective that performance is sometimes invariant to model architectures when employing BERT.

Another exciting area of overlap is in the use of graphical models to assist with text classification. Sometimes graph neural networks (or GNNs) are used, but another useful employment of graphs is in maintaining knowledge graphs, which can model semantic relations excellently.

4 Future work

We identify three possible different avenues of further work from these articles. The first is to create a classification model specifically designed to handle smaller datasets and classify their sentiment. This could be useful because reviews are given about a

wide variety of different goods and services that can sometimes have extremely niche characteristics that a more broadly trained model may not generalize well to. This effort could involve establishing methods of finetuning, or possibly altering existing architectures or vectorizing techniques that support the text classification models.

Another idea that came up in different models is leaning more into the world of graphical models. The paper on embeddings in text classification points out that knowledge graphs and knowledge embeddings are effective and woefully underutilized tools that thus far have been underemployed as a result of their disjoint training from accompanying word embeddings. The possibility of training a word embedding and knowledge embedding system at the same time and by the same methods could lend itself to a more powerful and high performing embeding system for capturing semantic relationships in the process of performing text classification.

The last idea has the most nebulous relation to the articles explored here but remains the most tangibly exciting. Having reviewed such a sprawling array of papers and surveys within text classification, we have seen a number of extremely varied and effective methods of text classification. With this arsenal of tools, we may attempt to tackle the problem of distinguishing between AI generated text and human generated text, which is an extremely topical and interesting challenge.

References

[1] Markus Bayer, Marc-André Kaufhold, and Christian Reuter. 2022. A survey on data augmentation for text classification. *ACM Computing Surveys*, 55(7):1–39.

[2] Xiaolong Chen, Jieren Cheng, Jingxin Liu, Wenghang Xu, Shuai Hua, Zhu Tang, and Victor S. Sheng. 2022. A survey of multi-label text classification based on deep learning. In *Artificial Intelligence and Security: 8th International Conference, ICAIS 2022, Qinghai, China, July 15–20, 2022, Proceedings, Part I*, page 443–456, Berlin, Heidelberg. Springer-Verlag.

[3] Liliane Costa, Italo Oliveira, and Renato Fileto. 2023. Text classification using embeddings: a survey. *Knowledge and Information Systems*, pages 1–43.

[4] M Mercy Evangeline and K Shyamala. 2021. Text categorization techniques: A survey. In *2021 International Conference on Innovative Practices in Technology and Management (ICIPTM)*, pages 137–142.

[5] Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S. Yu, and Lifang He. 2020. A survey on text classification: From shallow to deep learning. *CoRR*, abs/2008.00364.

[6] Abha Kiran Rajpoot, Parma Nand, and Ali Imam Abidi. 2021. A comprehensive survey on effective feature selection approaches for text sentiment classification process. *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 971–977.

[7] Pramod Sunagar, Anita Kanavalli, and S Shweta. 2020. A survey report on hypernym techniques for text classification. In *2020 Fifth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pages 65–70.

[8] Kunze Wang, Yihao Ding, and Soyeon Caren Han. 2023. Graph neural networks for text classification: A survey.

[9] Linzhong Xia, Dean Luo, Chunxiao Zhang, and Zhou Wu. 2019. A survey of topic models in text classification. In *2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pages 244–250.