A Quantum-Inspired Sentiment Representation Model

Harsh Vyas, Birva Oza, Darshit Kalariya and Parshwa Dand

Abstract

Sentiment analysis aims to capture sentiment information expressed in natural language texts and has been a critical research topic in artificial intelligence. The recently proposed Quantum-Inspired Sentiment Representation (QSR) model offers a novel approach by integrating sentiment information into document representations using density matrices constructed from sentiment phrases and semantic features. The original study demonstrated strong performance on widely used datasets such as the Obama-McCain Debate (OMD) dataset and the Sentiment140 dataset, significantly outperforming state-of-the-art baselines. In this reproducibility study, we implemented the QSR model and evaluated its performance under the same experimental settings. While our reproduced results align closely with the original in some tasks, we observed notable deviations in others, suggesting potential challenges in implementation details or dataset preprocessing. This study highlights the effectiveness of the QSR model while emphasizing the complexities of achieving reproducibility in advanced machine-learning approaches.

Keywords

Sentiment Analysis, Sentiment Representation, Quantum Theory, Density Matrix, QSR,

1. Problem

Sentiment analysis has long been a central task in natural language processing (NLP), with the goal of automatically determining the sentiment expressed in textual data. Traditional approaches to sentiment analysis focus primarily on capturing the semantic and contextual aspects of language. Methods such as term frequency-inverse document frequency (TF-IDF), n-grams, and word embeddings have been widely used to model the textual information of documents. Although these techniques effectively capture the syntactic and semantic relationships between words, they often overlook sentiment information that is critical for tasks such as sentiment classification, opinion mining, and emotional tone detection.

A significant limitation of many existing sentiment analysis models is their inability to integrate sentiment and semantic information in a unified framework. In other words, while some models can learn semantic similarities between words (e.g., “beautiful” and “cute”), they fail to recognize the shared sentiment polarity these words represent (for example, both are strongly positive).Given these challenges, it becomes clear that learning sentiment information is as crucial as learning semantic features to achieve high-quality sentiment classification.

To address these issues, the paper proposes a novel approach: the Quantum-Inspired Sentiment Representation (QSR)[1] model. This model aims to capture both the sentiment and semantic aspects of text in a unified manner. Using quantum probability theory, we model documents as density matrices derived from sentiment and semantic projectors. These projectors represent individual words and sentiment phrases extracted through part-of-speech tagging, focusing on adjectives and adverbs, which are widely recognized as strong indicators of sentiment. The QSR model offers a probabilistic framework in which each document is treated as a sequence of events (projectors) and the resulting density matrix captures the probability distribution over these events. This approach not only captures the semantic relationships between words, but also embeds the sentiment information directly into the document representation.

Introduction to Information Retrieval (IT550), Autumn 2024, DAIICT, India

202311015 (H. Vyas); 202311050 (B. Oza); 202311035 (D. Kalariya); 202311016 (P. Dand)

In this paper[2], we present the QSR model, evaluate it on popular Twitter sentiment analysis datasets, and compare its performance to existing baselines. We show that our model outperforms traditional approaches by integrating sentiment information into the learning process, offering a more holistic representation of text for sentiment classification.

2. Dataset

The study employed two prominent datasets for sentiment analysis. The **Sentiment140 dataset**, a widely used resource, contains 1.56 million tweets balanced between positive and negative sentiments. For the experiments conducted in this analysis, a smaller, randomly selected subset was utilized, comprising **1560 positive** and **1501 negative** tweets. This subset was chosen to maintain computational efficiency while preserving the dataset's representativeness.

The second dataset, the **Strict OMD dataset**, is a refined version of the Obama-McCain Debate dataset. This dataset includes tweets collected during the first U.S. presidential TV debate, featuring **509 tweets labeled as positive** and **741 as negative**. The Strict OMD dataset was chosen for its ability to provide more context-specific sentiment analysis, as the tweets are tied to a particular event and exhibit nuanced language patterns.

Both datasets underwent preprocessing steps, including **text cleaning**, spelling corrections, removal of **null and empty values**, and elimination of **stop words and punctuation** to ensure data quality.

3. Evaluation Metrics

The evaluation framework employed a comprehensive set of metrics to gauge model performance. **Accuracy**, a primary metric, assessed the overall proportion of tweets correctly classified into positive or negative sentiment categories, providing an overarching view of the model's effectiveness. **Precision** measured the proportion of correctly predicted positive tweets out of all tweets predicted as positive, focusing on the quality of positive predictions. **Recall** evaluated the model's capability to identify all true positive tweets within the dataset, emphasizing its sensitivity. The **F1 score**, calculated as the harmonic mean of precision and recall, served as a balanced metric, particularly valuable for datasets with class imbalances.

To ensure a robust evaluation, the study adopted a **5-fold cross-validation** technique. This method divided each dataset into five subsets, with one subset used for testing and the remaining four for training in each iteration. By cycling through all subsets, the approach minimized biases introduced by specific train-test splits. The average performance metrics across all folds provided a reliable and generalizable measure of the model's effectiveness. This rigorous evaluation ensured the reliability of the findings and facilitated fair comparisons across different models and datasets.

4. Results

The following tables compares the performance of various classifiers and algorithms on the Sentiment140 and Strict OMD datasets.

Table 1
Performance Metrics for Sentiment140 Dataset

Classifier	Algorithm	Accuracy		Precision		Recall		F1 Score	
		Paper	Our Results	Paper	Our Results	Paper	Our Results	Paper	Our Results
NB	Unigram	0.5581	0.5368	0.5626	0.5371	0.5539	0.5338	0.5567	0.5354
	Bigram	0.5564	0.5077	0.5680	0.5136	0.5561	0.5115	0.5623	0.5125
	Trigram	0.5546	0.5073	0.6009	0.5123	0.5326	0.5108	0.5762	0.5116
	Doc2vector	0.5548	0.5897	0.5446	0.5929	0.6483	0.5872	0.5904	0.5900
	Our QSR model	**0.5669**	**0.5485**	**0.5698**	**0.5495**	**0.5672**	**0.5492**	**0.5684**	**0.5494**
SVM	Unigram	0.5545	0.5682	0.5569	0.5682	0.5600	0.5679	0.5576	0.5680
	Bigram	0.5696	0.5031	0.5630	0.5023	0.5774	0.5023	0.5679	0.5023
	Trigram	0.5728	0.4940	0.5698	0.4923	0.5752	0.4925	0.5721	0.4924
	Doc2vector	0.5614	0.6473	0.5600	0.6483	0.6323	0.6461	0.5979	0.6473
	Our QSR model	**0.6567**	**0.5633**	**0.6492**	**0.5633**	**0.6548**	**0.5641**	**0.6526**	**0.5632**
RF	Unigram	0.5761	0.5721	0.5815	0.5763	0.5871	0.5697	0.5842	0.5730
	Bigram	0.5762	0.5165	0.5780	0.5162	0.6097	0.5153	0.5934	0.5158
	Trigram	0.5516	0.4900	0.5577	0.4859	0.5613	0.4877	0.5595	0.4868
	Doc2vector	0.5548	0.6345	0.5565	0.6362	0.6032	0.6335	0.5789	0.6349
	Our QSR model	**0.6283**	**0.5606**	**0.6204**	**0.5621**	**0.6678**	**0.5586**	**0.6432**	**0.5604**
Classfication Algorithm	SentiWordNet	0.5111	0.5734	0.5132	0.5735	0.5149	0.5735	0.5144	0.5735
	PMI-IR	0.5205	0.5639	0.5268	0.5774	0.5387	0.5593	0.5327	0.5682
	SentiStrength	0.5436	0.6381	0.5486	0.6757	0.5444	0.6334	0.5469	0.6539
CNN	CNN-Word Embedding	0.7008	0.8532	0.7033	0.8572	0.6957	0.8532	0.6993	0.8552
	CNN QSR	**0.7185**	**0.6483**	**0.7108**	**0.6352**	**0.7067**	**0.6254**	**0.7083**	**0.6303**
LSTM	Standard LSTM	0.6813	0.8215	0.6772	0.8277	0.6839	0.8225	0.6811	0.8251
	AT-LSTM	0.6929	0.8107	0.6968	0.8209	0.6922	0.8122	0.6948	0.8165
FCDNN	FCDNN-QSR	**0.5917**	**0.5581**	**0.5886**	**0.5622**	**0.5905**	**0.5662**	**0.5900**	**0.5642**

Table 2
Performance Metrics for Strict OMD Dataset

Classifier	Algorithm	Accuracy		Precision		Recall		F1 Score	
		Paper	Our Results	Paper	Our Results	Paper	Our Results	Paper	Our Results
NB	Unigram	0.6030	0.6791	0.5545	0.6527	0.5667	0.6421	0.5361	0.6474
	Bigram	0.5861	0.6441	0.5857	0.6126	0.6000	0.6015	0.5934	0.6070
	Trigram	0.5338	0.6398	0.5258	0.6065	0.6844	0.5911	0.5968	0.5987
	Doc2vector	0.5979	0.5186	0.5654	0.5437	0.5489	0.5470	0.5570	0.5453
	Our QSR model	**0.6477**	**0.6114**	**0.6471**	**0.6035**	**0.6367**	**0.6105**	**0.6432**	**0.6069**
SVM	Unigram	0.6231	0.7325	0.6161	0.7142	0.5575	0.7034	0.5835	0.7087
	Bigram	0.6164	0.5928	0.6217	0.5445	0.6267	0.5332	0.6198	0.5388
	Trigram	0.6224	0.5721	0.5831	0.5045	0.5844	0.5036	0.5816	0.5040
	Doc2vector	0.6231	0.7249	0.5772	0.7032	0.5225	0.6939	0.5482	0.6985
	Our QSR model	**0.6528**	**0.6561**	**0.6533**	**0.6316**	**0.6520**	**0.6311**	**0.6526**	**0.6313**
RF	Unigram	0.6231	0.7271	0.6167	0.7314	0.6111	0.6722	0.6219	0.7005
	Bigram	0.6047	0.6474	0.6105	0.6602	0.6122	0.5527	0.5976	0.6011
	Trigram	0.5829	0.5605	0.5854	0.5961	0.5878	0.5298	0.5832	0.5605
	Doc2vector	0.6130	0.6452	0.6165	0.6517	0.6128	0.5585	0.6171	0.6007
	Our QSR model	**0.6450**	**0.6604**	**0.6450**	**0.7100**	**0.6390**	**0.5676**	**0.6411**	**0.6301**
Classfication Algorithm	SentiWordNet	0.5335	0.5545	0.5361	0.5477	0.5380	0.5505	0.5373	0.5491
	PMI-IR	0.5846	0.4247	0.5856	0.6061	0.5844	0.5302	0.5876	0.5656
	SentiStrength	0.5776	0.6135	0.5800	0.6806	0.5765	0.6670	0.5782	0.6738
CNN	CNN-Word Embedding	0.7724	0.8919	0.7693	0.8803	0.7751	0.8974	0.7744	0.8887
	CNN QSR	**0.7775**	**0.6386**	**0.7808**	**0.6105**	**0.7747**	**0.6071**	**0.7759**	**0.6088**
LSTM	Standard LSTM	0.7710	0.8679	0.7565	0.8561	0.7731	0.8633	0.7676	0.8597
	AT-LSTM	0.7806	0.7958	0.7768	0.7796	0.7822	0.7840	0.7784	0.7818
FCDNN	FCDNN-QSR	**0.6550**	**0.6583**	**0.6618**	**0.6422**	**0.6522**	**0.6396**	**0.6561**	**0.6409**

5. Key Problems and Learning

Several challenges emerged, highlighting the inherent difficulties in replicating complex models with limited methodological details provided in the original paper. The paper mentions taking a random sample of approximately 3,000 tweets from the Sentiment140 dataset, which contains 1.6 million records, but it does not specify the sampling criteria or process. As a result, we also had to randomly sample the dataset, leading to slight variations in the dataset composition. Similarly, for the Obama-McCain Debate (OMD) dataset, we utilized a strict version with fewer rows than the one presumably used in the paper. These variations in dataset size and sampling processes introduced differences in performance metrics such as accuracy, precision, recall, and F1-score, with observed deviations of ±5% to ±10%. At times, our results were better than those reported in the paper, while in other instances, they fell short.

Another key challenge was the lack of critical implementation details. While the QSR model relies on advanced mathematical constructs, such as representing words and phrases as projectors encapsulated in density matrices through maximum likelihood estimation, the paper does not provide specific information on initializing weights, or adjusting parameters like the step size t_k. These omissions forced us to make assumptions and estimations, which likely contributed to deviations in our results. Implementing the QSR model from scratch also required translating theoretical formulations into practical code. Given the complexity of quantum probability theory and the absence of an open-source implementation, minor discrepancies in mathematical operations or assumptions could have compounded into noticeable performance differences. Moreover, while the original paper demonstrated consistent improvements across various classifiers when using QSR, our implementation showed lower results compared to the reported outcomes, suggesting that certain nuances or optimizations in their approach were not effectively captured in ours.

The preprocessing steps further compounded these challenges. Sentiment phrases, a critical component of the QSR model, were identified using part-of-speech tagging to extract adjectives and adverbs. Differences in tokenization methods, part-of-speech tagging tools, or preprocessing pipelines could have influenced the model's outcomes. Through this project, we learned the critical importance of detailed methodological documentation for reproducibility, especially regarding dataset preparation, parameter initialization, and hyperparameter adjustments. Implementing a mathematically complex model like QSR from scratch deepened our understanding of quantum-inspired representations and highlighted the sensitivity of such models to subtle changes. Additionally, this task enhanced our skills in debugging, managing machine learning pipelines, and addressing reproducibility challenges in research.

References

[1] Y. Zhang, D. Song, P. Zhang, X. Li, P. Wang, A quantum-inspired sentiment representation model for twitter sentiment analysis, Applied Intelligence 49 (2019) 3093–3108.

[2] I. Basile, F. Tamburini, Towards quantum language models, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1840–1849.