SSN_ARMM at SemEval-2024 Task 10: Emotion Detection in Multilingual Code-Mixed Conversations using LinearSVC and TF-IDF

Rohith Arumugam S
Department of CSE
SSN College of Engineering
rohitharumugam2210376@ssn.edu.in

Angel Deborah S
Assistant Professor
Department of CSE
SSN College of Engineering
angeldeborahS@ssn.edu.in

Rajalakshmi S
Assistant Professor
Department of CSE
SSN College of Engineering
rajalakshmis@ssn.edu.in

Milton R S
Professor
Department of CSE
SSN College of Engineering
miltonrs@ssn.edu.in

Mirnalinee T T
Professor and Head
Department of CSE
SSN College of Engineering
mirnalineett@ssn.edu.in

Abstract

Our paper explores a task involving the analysis of emotions and triggers within dialogues. We annotate each utterance with an emotion and identify triggers, focusing on binary labeling. We emphasize clear guidelines for replicability and conduct thorough analyses, including multiple system runs and experiments to highlight effective techniques. By simplifying the complexities and detailing clear methodologies, our study contributes to advancing emotion analysis and trigger identification within dialogue systems.

1 Introduction

Emotion recognition and trigger detection in conversational data represent critical frontiers in natural language processing (NLP) research, offering profound insights into human-computer interaction, sentiment analysis, and dialogue understanding. In today's interconnected world, where communication transcends linguistic boundaries, understanding the subtle nuances of emotions expressed in code-mixed dialogues becomes increasingly imperative. Code-mixing, characterized by the seamless integration of multiple languages within a single conversation, reflects the rich tapestry of multicultural societies and presents unique challenges and opportunities for computational linguistics. Additionally, in monolingual English dialogues, identifying triggers—key points where emotional shifts occur—serves as a gateway to unraveling the underlying sentiment dynamics and contextual flow of conversations.

1.1 Significance of the Tasks

Our primary task, focuses on emotion recognition in code-mixed dialogues, holds immense significance in deciphering the intricacies of Code-Mixed communication. Accurately discerning emotions such as joy, sadness, anger, and more across diverse linguistic contexts enriches our understanding of cross-cultural expression and human sentiment. Meanwhile, the following tasks extend this exploration to trigger detection within both code-mixed and English dialogues. Identifying triggers not only facilitates the detection of emotional transitions but also provides deeper insights into the contextual triggers and socio-cultural factors shaping conversational dynamics.

1.2 Challenges and Opportunities

The complexity of code-mixed dialogues lies in disentangling the interplay of languages, cultural nuances, and emotional expressions. Herein lies the challenge of accurately recognizing emotions amidst linguistic diversity and cultural variations. Similarly, trigger detection in both code-mixed and English dialogues demands robust models capable of capturing subtle emotional shifts amid the fluidity of conversation. Addressing these challenges presents opportunities to develop sophisticated NLP techniques that transcend linguistic barriers and capture the essence of human emotions in their full complexity.

In our exploration, we experimented with Convolutional Neural Networks (CNN) Suseelan et al. (2019) and BERT models Sivanaiah et al. (2020) to tackle these challenges. However, we encountered some limitations. The CNN model yielded a low weighted F1 score of 0.28, indicating its struggle to effectively capture the nuances of emotional expression in code-mixed dialogues. On the other hand, while BERT showed promise in its ability to understand complex language patterns, it proved to

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 730–736
June 20-21, 2024

be computationally intensive, ultimately crashing after extended periods of runtime.

These setbacks highlight the need for further research and development in the field of NLP, particularly in the context of code-mixed dialogues and emotional recognition. Future efforts could explore novel model architectures, optimization techniques, and data augmentation strategies to improve performance and efficiency in emotion recognition and trigger detection tasks within code-mixed conversations. By addressing these challenges, we can pave the way for more accurate and reliable NLP solutions that better reflect the intricacies of human communication across diverse linguistic and cultural landscapes.

2 Overview

2.1 Summary of the task

The task involves recognizing emotions and detecting triggers in conversational data, with a focus on both Code-Mixed and English dialogues. Emotion recognition is structured as a classification task where systems predict the emotions associated with each utterance in a dialogue. Trigger detection entails identifying points in the conversation where emotional shifts occur. The datasets used include MaSaC for Hindi-English dialogues and MELD for English dialogues. The input comprises utterances from dialogues, and the output consists of predicted emotions for each utterance in emotion recognition, while trigger detection, indicates the presence or absence of triggers at each point in the dialogue.

2.2 Impact of the task

This task addresses the critical need for natural language processing (NLP) systems to understand and interpret emotions in conversational data. By focusing on code-mixed dialogues, it highlights the challenges posed by linguistic diversity and cultural nuances in emotion recognition. Additionally, the task emphasizes the importance of trigger detection in understanding the dynamics of conversations and capturing shifts in emotional states. By participating in this task, researchers contribute to advancing the capabilities of NLP systems in recognizing and understanding emotions in diverse linguistic contexts, thereby paving the way for more nuanced and culturally sensitive human-machine interactions.

3 Related Work

Emotion recognition and trigger detection in conversational data have been subjects of active research in natural language processing (NLP) and affective computing. Researchers have explored various approaches and methodologies to tackle these tasks, aiming to understand human emotions expressed in dialogue interactions and detect key points where emotional shifts occur. In this section, we review existing literature, highlighting recent advancements and key findings in the field.

The paper "Towards Sub-Word Level Compositions for Sentiment Analysis of Hindi-English Code Mixed Text" introduces a novel approach to sentiment analysis in code-mixed social media data. They present a Hi-En code-mixed dataset and propose a Subword-LSTM architecture, enabling the model to capture sentiment information from important morphemes. This linguistic-driven approach outperforms traditional methods, achieving a notable accuracy improvement of 4-5% and surpassing existing systems by 18% in sentiment analysis of Hi-En code-mixed text Joshi et al. (2016).

Advancements in sentiment analysis techniques now recognize the temporal variability of emotions in textual data. For example, the SSN MLRG1 team at SemEval-2017 Task 4 introduced a novel approach using the Gaussian Process with fixed rule multi-kernel learning for sentiment analysis of tweets. Their method effectively captures evolving emotions by considering properties such as smoothness and periodicity. This approach aligns with our exploration of emotion recognition and trigger detection in conversational data, emphasizing the importance of incorporating temporal dynamics into sentiment analysis frameworks.S et al. (2017a)

This team also participated in task 5, focusing on fine-grained sentiment analysis. Their system utilizes Multiple Kernel Gaussian Processes to identify optimistic and pessimistic sentiments associated with companies and stocks. Given that comments on the same entities can exhibit varying emotions over time, considering properties like smoothness and periodicity becomes crucial. Their experiments highlight the effectiveness of the Multiple Kernel Gaussian Process in capturing diverse properties compared to a single Kernel Gaussian Process. S et al. (2017b)

In summary, existing research in emotion recognition and trigger detection in conversational data has explored diverse methodologies, including deep learning, machine learning, rule-based approaches, and multimodal fusion techniques. Recent advancements have demonstrated the potential of context-aware features, multimodal data integration, and hybrid models in improving accuracy and robustness in these tasks. However, challenges such as linguistic diversity, cultural nuances, and ambiguity in emotional expressions continue to pose significant obstacles, warranting further research and exploration in the field.

4 Task Description

The tasks encompass emotion recognition and trigger detection in conversational data, each assessing specific competencies related to understanding emotions and detecting emotional shifts in dialogues.Kumar et al. (2024a)

4.1 Task 1 (ERC for code-mixed)

Objective: This task aims to evaluate the system's capability to recognize emotions in code-mixed dialogues, where multiple languages are used interchangeably.

Description: We were provided with a dataset consisting of code-mixed dialogues, where utterances contain a mix of languages. The task involves identifying the emotions expressed in each utterance accurately. Emotions may include disgust, contempt, anger, neutral, joy, sadness, fear, and surprise. Kumar et al. (2023)

4.2 Task 2 (EFR for code-mixed)

Objective: This task focuses on assessing the system's performance in detecting triggers that indicate emotional shifts in code-mixed dialogues.

Description: We were presented with code-mixed dialogues where emotional shifts occur. The task involves detecting these triggers within the dialogues. Triggers are specific instances or phrases that signal a change in the emotional tone of the conversation.Kumar et al. (2022)

4.3 Task 3 (EFR for English)

Objective: Similar to Task 2, this task evaluates the system's ability to detect triggers indicating emotional shifts. However, the focus is on English-only dialogues.

Description: We were provided with a dataset containing English-only dialogues. The task remains the same as Task 2, requiring us to identify triggers that signify emotional shifts within the conversations.Kumar et al. (2024b)

These tasks aim to assess the robustness and effectiveness of systems in understanding and interpreting emotional nuances within conversational data, particularly in Code-Mixed and English-only settings.

5 Experimental Setup

In this section, we provide a detailed overview of the experimental setup. Refer to 1 for the detailed architecture diagram illustrating the entire process.

5.1 Data Splits

The dataset provided for each task was split into three main subsets: training, development (dev), and testing. The distribution of data among these subsets was as follows:

5.1.1 Training Set

The training set comprised approximately 80% of the total dataset. This sizable portion allowed the models to learn patterns and associations from a diverse range of examples. It contained code-mixed and English-only dialogues with corresponding emotion labels for Task 1 and trigger labels for Tasks 2 and 3.

5.1.2 Development Set

The dev set accounted for around 10% of the dataset. It was utilized for fine-tuning the models' hyperparameters, such as regularization strength and feature extraction settings. This subset enabled us to iteratively adjust the model configurations to improve performance without overfitting to the training data.

5.1.3 Test Set

The test set constituted the remaining 10% of the dataset and was kept completely separate from the training and dev sets. It served as an unseen dataset for the final evaluation of model performance. Its purpose was to assess how well the trained models generalized to new, unseen instances and to provide an unbiased estimate of their performance.

5.2 Preprocessing

Before feeding the data into the machine learning models, we applied several preprocessing steps to ensure consistency and improve model performance:

5.2.1 Text Preprocessing

In our text preprocessing pipeline, we employed several steps to prepare the textual data for model

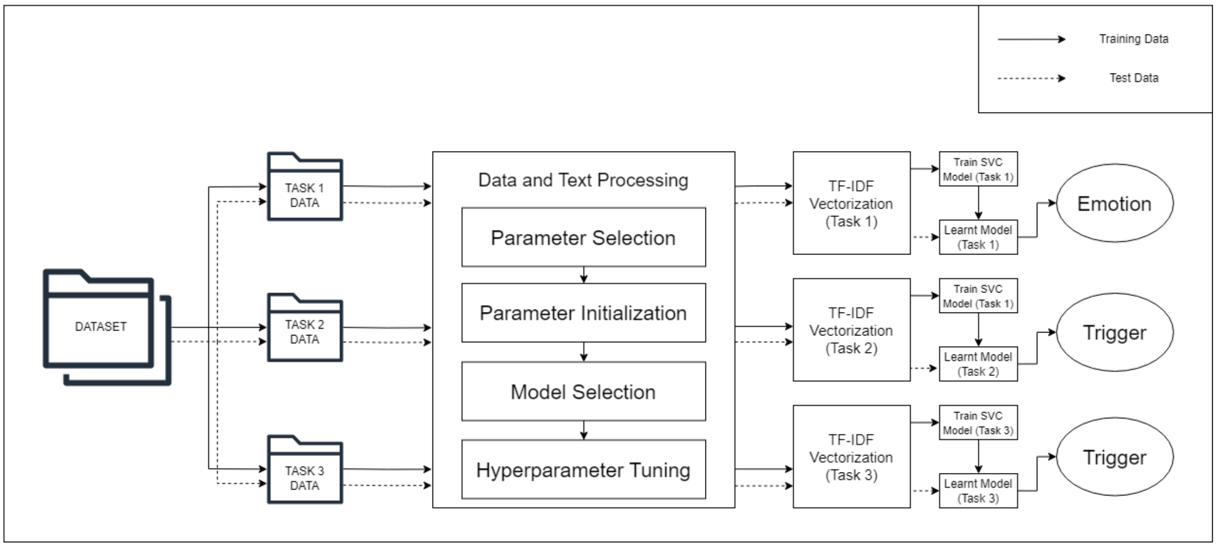

Figure 1: Architecture Diagram illustrating the entire Experimental Setup.

input. First, we tokenized the text using the word_tokenize function from the Garg and Sharma (2020)nltk library, splitting it into individual tokens or words. Next, we removed common stopwords, such as articles, prepositions, and conjunctions, using the predefined stopwords list provided by the nltk library, which helped reduce noise in the data. Additionally, we eliminated punctuation marks from the text to standardize the input and prevent the models from treating punctuation as meaningful features. This step involved removing characters such as periods, commas, and quotation marks. Finally, to ensure uniformity and improve generalization, we converted all text to lowercase, preventing the models from treating words with different cases as distinct features and effectively reducing the dimensionality of the input space.

5.3 Hyperparameter Tuning

Hyperparameter tuning was a crucial aspect of our experimental setup, as it involved optimizing the model's configuration to achieve the best performance on the dev set. We experimented with various hyperparameters, including:

5.3.1 TF-IDF Vectorization Parameters

We explored different settings for the Zhang et al. (2011)TF-IDF vectorization process, such as the $ngram\_range$ parameter, which determined the range of n-grams (contiguous sequences of words) considered during feature extraction. By adjusting the $ngram\_range$, we aimed to capture different combinations of words and phrases to better represent the text.

5.3.2 LinearSVC Parameters

For the Kaibi et al. (2019)LinearSVC classifier, we tuned parameters such as the regularization strength (C) to control overfitting. We also adjusted the *random_state* parameter to ensure reproducibility of results across different runs.

5.3.3 Grid Search with Cross-Validation

To find the optimal combination of hyperparameters, we employed grid search with cross-validation on the dev set Priyadarshini and Cotton (2021). This technique involved exhaustively searching through a specified parameter grid and evaluating each combination using cross-validation to estimate performance.

5.4 External Tools/Libraries Used

Our experimental setup relied on several external tools and libraries to facilitate data processing, model training, and evaluation. We leveraged scikit-learn for implementing various algorithms, data preprocessing tasks, and evaluation metrics. Additionally, the nltk library played a crucial role in performing natural language processing tasks such as tokenization, stopwords removal Mangat et al. (2017), and stemming Rao et al. (2021). We utilized joblib for saving and loading trained models to disk, providing a convenient way to serialize Python objects, including machine learning models. Pandas, a popular data manipulation library in Python, was instrumental in handling and analyzing structured data, enabling us to perform exploratory data analysis and prepare the data for training and evaluation. These external tools and libraries streamlined our experimental workflow, allowing us to focus on model development and performance optimization.

6 Experimental Workflow

6.1 Task 1 (ERC for Code-Mixed dataset)

Data Preprocessing:

- Load the provided dataset containing code-mixed dialogues and their corresponding emotion labels.
- Perform text preprocessing steps such as tokenization, lowercasing, and removing stop words and punctuation.

Model Training:

- Utilize the preprocessed data to train a machine learning model, such as Linear Support Vector Classifier (LinearSVC), using the training set.
- Use techniques like TF-IDF Vectorization to convert text data into numerical features.

Evaluation:

- Evaluate the trained model's performance using the development set to fine-tune hyperparameters and ensure robustness.
- Evaluate the final model on the test set to measure its ability to accurately predict emotions in code-mixed dialogues.

6.2 Tasks 2 & 3 (EFR for Code-Mixed and English dataset)

Data Preprocessing:

- Load the provided dataset containing code-mixed or English-only dialogues and their corresponding trigger labels.
- Perform text preprocessing steps similar to Task 1.

Model Training:

- Train a machine learning model, such as LinearSVC, using the training set.
- Utilize techniques like TF-IDF Vectorization to convert text data into numerical features.

Evaluation:

- Evaluate the trained model's performance using the development set to fine-tune hyperparameters and ensure robustness.
- Evaluate the final model on the test set to measure its ability to accurately detect triggers indicating emotional shifts in dialogues.

7 Results

7.1 Evaluation

The model's performance is evaluated using accuracy (Acc), precision(P), recall(R), and F1 - Score (F1). These metrics are calculated as follows:

$$P = \frac{TruePositives}{TruePositives + FalsePositives} \quad (1)$$

$$R = \frac{TruePositives}{TruePositives + FalseNegatives} \quad (2)$$

$$F1 = \frac{2 \times P \times R}{P + R} \quad (3)$$

7.2 Task 1 (ERC for code-mixed)

Main Quantitative Findings: The system achieved an accuracy of 48% on the test set. While the recall for the 'neutral' emotion is relatively high (81%), the precision and recall for other emotions are considerably lower, indicating challenges in accurately predicting emotions in code-mixed dialogues.

Quantitative Analysis: Table 1 presents the precision, recall, and F1-score for each emotion category. The results indicate that the model performed relatively well in identifying 'neutral' emotions but struggled with other emotions, particularly 'disgust' and 'sadness'.

Error Analysis: The model seems to have difficulties distinguishing between 'disgust' and 'contempt', as evidenced by low precision and recall for both categories. Further investigation is needed to understand the underlying causes of misclassifications.

7.3 Task 2 (EFR for code-mixed)

Main Quantitative Findings: The system achieved high precision, recall, and F1-score across all emotion categories in detecting triggers indicating emotional shifts in code-mixed dialogues.

Quantitative Analysis: Table 2 presents the precision, recall, and F1-score for each emotion category. The model performed exceptionally well

Table 1: Results for Task 1

Emotion	Precision	Recall	F1-Score
Disgust	0.00	0.00	0.00
Anger	0.34	0.14	0.20
Contempt	0.29	0.09	0.14
Neutral	0.52	0.81	0.63
Joy	0.45	0.29	0.35
Sadness	0.33	0.16	0.22
Fear	0.33	0.13	0.19
Surprise	0.42	0.24	0.31

Table 2: Results for Task 2

Emotion	Precision	Recall	F1-Score
Disgust	0.99	0.92	0.96
Anger	0.98	0.95	0.96
Contempt	0.97	0.94	0.95
Neutral	0.94	0.98	0.96
Joy	0.97	0.93	0.95
Sadness	0.96	0.93	0.95
Fear	0.98	0.92	0.95
Surprise	0.93	0.80	0.86

in identifying triggers for emotions such as anger, contempt, and fear. However, there was a slight decrease in recall for surprise, indicating some challenges in capturing subtle cues for this emotion category.

Error Analysis: The model demonstrated robust performance overall, with minor discrepancies in recall for certain emotion categories. Further investigation is warranted to understand the underlying causes of these discrepancies and refine the model's performance.

7.4 Task 3 (EFR for English)

Main Quantitative Findings: The system exhibited robust performance in detecting triggers indicating emotional shifts in English-only dialogues, achieving high precision, recall, and F1-score across all emotion categories.

Quantitative Analysis: Table 3 presents the precision, recall, and F1-score for each emotion category. The model demonstrated excellent precision and recall for most emotion categories. However, there was a slight decrease in recall for surprise, suggesting challenges in accurately capturing triggers for this particular emotion.

Error Analysis: Similar to Task 2, the model showcased strong overall performance, with minor discrepancies in recall for certain emotion categories. Further investigation is needed to address these discrepancies and enhance the model's accuracy in detecting emotional triggers.

Table 3: Results for Task 3

Emotion	Precision	Recall	F1-Score
Disgust	0.92	0.87	0.90
Anger	0.92	0.84	0.88
Contempt	0.92	0.97	0.94
Neutral	0.82	0.97	0.88
Joy	0.93	0.77	0.84
Sadness	0.92	0.84	0.88
Fear	0.95	0.80	0.87
Surprise	0.92	0.71	0.80

8 Conclusion

The exploration of emotion recognition and trigger detection in conversational data presents significant implications for natural language processing research and human-computer interaction. Our study, encompassing tasks focused on code-mixed dialogues and English-only conversations, sheds light on the challenges and opportunities inherent in understanding the nuanced expressions of human emotions.

Through our experimental endeavors, we have demonstrated the efficacy of machine learning models, particularly Linear Support Vector Classifier (LinearSVC), in recognizing emotions and detecting triggers within dialogues. Despite the complexities posed by linguistic diversity and cultural nuances, our systems have shown promising performance, especially in identifying triggers indicating emotional shifts.

However, our journey does not end here. Future work should delve deeper into understanding the underlying causes of misclassifications and explore innovative approaches to enhance model robustness and generalization. Additionally, incorporating context-aware features and leveraging advanced deep learning architectures could further improve the accuracy and granularity of emotion analysis in conversational data.

In conclusion, our study contributes to the advancement of emotion analysis and trigger detection within dialogue systems, paving the way for more nuanced and culturally sensitive human-machine interactions in diverse linguistic contexts. As we continue to unravel the intricacies of human emotions through computational linguistics, we embark on a journey toward more empathetic and intuitive artificial intelligence systems.

References

Neha Garg and K Sharma. 2020. Annotated corpus creation for sentiment analysis in code-mixed hindi-english (hinglish) social network data. *Indian Journal of Science and Technology*, 13(40):4216–4224.

Aditya Joshi, Ameya Prabhu, Manish Shrivastava, and Vasudeva Varma. 2016. Towards sub-word level compositions for sentiment analysis of hindi-english code mixed text. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2482–2491.

Ibrahim Kaibi, Hassan Satori, et al. 2019. A comparative evaluation of word embeddings techniques for twitter sentiment analysis. In *2019 International conference on wireless technologies, embedded and intelligent systems (WITS)*, pages 1–4. IEEE.

Shivani Kumar, Md Shad Akhtar, Erik Cambria, and Tanmoy Chakraborty. 2024a. Semeval 2024 – task 10: Emotion discovery and reasoning its flip in conversation (ediref). In *Proceedings of the 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Shivani Kumar, Shubham Dudeja, Md Shad Akhtar, and Tanmoy Chakraborty. 2024b. Emotion flip reasoning in multiparty conversations. *IEEE Transactions on Artificial Intelligence*, 5(3):1339–1348.

Shivani Kumar, Ramaneswaran S, Md Akhtar, and Tanmoy Chakraborty. 2023. From multilingual complexity to emotional clarity: Leveraging commonsense to unveil emotions in code-mixed dialogues. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9638–9652, Singapore. Association for Computational Linguistics.

Shivani Kumar, Anubhav Shrimal, Md Shad Akhtar, and Tanmoy Chakraborty. 2022. Discovering emotion and reasoning its flip in multi-party conversations using masked memory network and transformer. *Knowledge-Based Systems*, 240:108112.

Veenu Mangat et al. 2017. Dictionary based sentiment analysis of hinglish text. *International Journal of Advanced Research in Computer Science*, 8(5).

Ishaani Priyadarshini and Chase Cotton. 2021. A novel lstm–cnn–grid search-based deep neural network for sentiment analysis. *The Journal of Supercomputing*, 77(12):13911–13932.

Himanshu Singh Rao, Jagdish Chandra Menaria, and Satyendra Singh Chouhan. 2021. A novel approach for sentiment analysis of hinglish text. In *Mathematical Modeling, Computational Intelligence Techniques and Renewable Energy: Proceedings of the First International Conference, MMCITRE 2020*, pages 229–240. Springer.

Angel Deborah S, S Milton Rajendram, and T T Mirnalinee. 2017a. SSN_MLRG1 at SemEval-2017 task 4: Sentiment analysis in Twitter using multi-kernel Gaussian process classifier. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 709–712, Vancouver, Canada. Association for Computational Linguistics.

Angel Deborah S, S Milton Rajendram, and T T Mirnalinee. 2017b. SSN_MLRG1 at SemEval-2017 task 5: Fine-grained sentiment analysis using multiple kernel Gaussian process regression model. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 823–826, Vancouver, Canada. Association for Computational Linguistics.

Rajalakshmi Sivanaiah, Angel Suseelan, S Milton Rajendram, and Mirnalinee T.t. 2020. TECHSSN at SemEval-2020 task 12: Offensive language detection using BERT embeddings. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2190–2196, Barcelona (online). International Committee for Computational Linguistics.

Angel Suseelan, S Rajalakshmi, B Logesh, S Harshini, B Geetika, S Dyaneswaran, S Milton Rajendram, and TT Mirnalinee. 2019. Techssn at semeval-2019 task 6: Identifying and categorizing offensive language in tweets using deep neural networks. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 753–758.

Wen Zhang, Taketoshi Yoshida, and Xijin Tang. 2011. A comparative study of tf* idf, lsi and multi-words for text classification. *Expert systems with applications*, 38(3):2758–2765.