Identifying Social Self in Text: A Machine Learning Study

Jaya Caporusso
jaya.caporusso@ijs.si
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

Matthew Purver
Jožef Stefan Institute
Ljubljana, Slovenia
Queen Mary University of London
London, UK

Senja Pollak
Jožef Stefan Institute
Ljubljana, Slovenia

Abstract

The Self encompasses many aspects, such as the Social Self. Identifying them in text is relevant for many purposes, including mental-health research. As part of a larger project aimed at automatically detecting Self-aspects in written language, in this study we annotate and employ a dataset of diary entries to classify the presence or absence of Social Self. We train three classifiers—Support Vector Machine (SVM), Naïve Bayes, and Logistic Regression—on either learned or predefined features. The best-performing model is the SVM trained on predefined LIWC features based on a previous study. We further apply feature importance methods, and examine which features make the biggest contribution to the classification models. The most informative feature across models trained on learned features is the word "we", while the LIWC category "social referents" emerges as the most important feature for models trained on predefined features.

Keywords

social self, machine learning, classification, feature importance

1 Introduction

A central aspect of human experience, the Self is a complex, multi-aspect phenomenon [3]. Its aspects—encompassing, for example, personal narratives [18] and social interactions [2]—correlate with other relevant constructs, such as mental-health conditions [17]. While the various Self-aspects reflect in the individual's language [14], Natural Language Processing (NLP) studies rarely explore them and employ them in-depth.

This work is part of a larger project aimed at developing models to automatically identify Self-aspects in text, with applications in mental-health-research and empirical phenomenology [5]. Due to the sensitive nature of the domains of application, we attempt an approach that allows both interpretability and ground-truth basis, opting for classical machine learning (ML) models. In this study, we focus on one Self-aspect: the Social Self (SS), defined as *the Self as it is shaped and/or perceived when in an interaction or relationship of sorts with other people or entities to whom we attribute qualities of inner life* [4]. We aim to investigate how this is represented in diary entries and whether these representations can be reliably identified using machine learning. Additionally, we explore which linguistic features are most predictive of these aspects. Identifying SS in text is valuable, as, e.g., disturbances in the SS are closely linked to mental health conditions [7]. This project involves labelling—with a mixed approach involving human annotators and large language models (LLMs)—diary entry instances as binary (representing or not) SS, with the purpose of investigating the correlation between SS and textual features. We train and compare three classifiers (i.e., Support Vector Machine (SVM), Naïve Bayes (NB), and Logistic Regression (LR)) to predict SS using either 1) learned features (i.e., TF-IDF unigrams and bigrams) or 2) predefined features (i.e., Linguistic Inquiry and Word Count (LIWC; [1]) lexicon categories (see [4]). We use the mentioned classifiers instead of LLMs (e.g., GPT-4) because our focus is on employing interpretable features and understanding their contribution to predictions—an aspect less directly accessible in generative models. We conduct feature importance analysis to explore these contributions further. The code is available at https://github.com/jayacaporusso/SELFtext upon request.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.sikdd.2

2 Related Work

Studies that address the correlation between text and the traits and states of the text's author often utilise the Linguistic Inquiry and Word Count (LIWC), a text analysis software developed to analyse linguistic and psychosocial constructs connected to various textual aspects [1] (e.g., [9]). Various studies have found Self states to be associated with linguistic features, e.g., depression with first-person singular pronouns [15]. This has been employed in classification tasks (e.g., [6]). In a previous study, after labelling a dataset with a mixed approach employing human annotation and LLMs, we analysed which LIWC-22 features characterise Reddit posts including Self as an Agent, Bodily Self, and SS [4]. Specifically, we showed that the **presence of SS** is correlated with LIWC categories including, among the others, *emotion* and *time related terms*. In contrast, the **absence of SS** is correlated with, e.g., *technology* and *negative emotions*. In this work, we employ this knowledge to build SS classifiers on predefined features and compare them with classifiers trained on learned features.

3 Research Questions

In this study, we aim to address the following main research questions (RQs). **RQ1**: How does a SS classifier trained on predefined features perform compared to a SS classifier trained on learned features? **RQ2**: Among the algorithms employed, which one performs better for our task? **RQ3**: Which features are more relevant for the classification of SS?

3.1 Labelling

In our study, we use a publicly available dataset in English [11] comprising 1,473 text samples (sub-entries; average length: 507.6 characters, 100.6 words) from 500 personal journal entries (500 anonymous subjects). We augment the dataset with binary labels for SS, as following addressed.

For labelling, we employ a mixed approach (see [4]) that combines human annotation with the large language model (LLM)

gemma2 [16]. The instructions for manual annotation are provided in the Appendix A. Two human annotators label the first 105 instances of the dataset. This is needed to calculate inter-annotator agreement with the LLM annotations. We instruct gemma2 to label the data three times, providing three different personalisations (see [10]): expert in phenomenology, cognitive psychology, or social psychology. Additionally, we provide them with definitions of SS, instructions to annotate it, examples of a text instance where it is present, a text instance where it is absent, and explanations of why this is so. These can be extracted from the instructions for manual annotation. Each gemma2 model performs a one-shot, binary classification for each self-aspect. We calculate majority voting with the resulting labels and compute the inter-annotator agreement between each pair among the human and the LLM annotators by calculating Cohen's Kappa coefficient. This results in Cohen's Kappa coefficients of 0.80 (human annotators), 0.89 (first annotator vs. gemma2), and 0.84 (second annotator vs. gemma2). In the further steps, we use the majority voting labels. The class balance (calculated on the majority voting) is 50.3% (SS present) vs 49.7% (SS not present).

4 Classification

The text is preprocessed, converting it to lowercase and removing punctuation and extra whitespace. We extract learned and predefined features. We then train three classifiers for each set of features: an SVM, a NB, and a LR model.

4.1 Feature Engineering

We are interested in comparing the performance of models trained on learned vs pre-defined features. In this study, we choose to employ TF-IDF calculated on unigrams and bigrams as learned features, and the LIWC features identified as being related to the presence or absence of SS in Caporusso et al. [4].

4.1.1 Learned Features. To extract learned features, we employ TfidfVectorizer, applying TF-IDF weighting to unigrams and bigrams. Restricting the representation to unigrams and bigrams, a common choice in exploratory text classification, efficiently displays feature importance, balancing interpretability and computational efficiency. We limit the feature space to the 1000 n-grams that, based on their TF-IDF scores, are the most informative. This ensures computational efficiency. In this process, we choose not to exclude stopping words. Indeed, for the purpose of our study, they do not merely constitute noise but might play a key role in distinguishing text instances reporting on SS.

4.1.2 Predefined Features. We analyse the presence of all the LIWC-22 [1] categories and subcategories, and subsequently only considered the LIWC features of interest. Specifically, as predefined features, we employ the LIWC features that Caporusso et al. [4] identified as being related to the presence and absence of SS (see 2), for example *authenticity*, *social referents*, and *the pronoun I*. For each of them, LIWC-22 provides scores relative to the text length. All LIWC features were standardised using Z-score normalisation to ensure comparability across different feature scales. This is particularly important for models like SVM and LR, which are sensitive to feature magnitudes. Missing values (NaNs) are handled using mean imputation.

4.2 Models

The models are trained and evaluated using 10-fold cross-validation to assess their performance. Specifically, we train three models on the learned features and three models on the predefined features. The models are of three different kinds: SVM, NB, and LR, all commonly used in text classification tasks. We employ default hyperparameters. For the SVM, we use Linear kernel. For LR, we apply L2 regularisation, which adds a penalty term to the model's objective function, minimising overfitting. For NB, MultinomialNB was used for learned features, while GaussianNB was used for predefined features, which consist of continuous numerical values derived from linguistic analysis. MultinomialNB assumes that features represent discrete frequency counts, while GaussianNB assumes that feature distributions follow a normal distribution, making it appropriate for continuous data.

5 Evaluation

Similarly to the training process, the models are evaluated using 10-fold cross-validation. All the models perform reasonably well, with the SVM model trained on predefined features outperforming them all (RQ1 and RQ2). The metrics (precision, recall, and F1-score: mean and STD) across folds are reported in Table 1. They match the macro average scores. The confusion matrices for each model are presented in Figures 3 and 4 in the Appendix B. These highlight that models trained on predefined features generally perform better at distinguishing between classes, with the SVM and LR models achieving higher accuracy for both Class 0 and Class 1. However, NB trained on predefined features struggles with a higher rate of false positives for Class 0. The models trained on learned features have slightly lower performance, with higher misclassification rates for Class 1 predictions. After performing a Friedman test across folds (statistic = 44.26, p-value = 0.00), we find a statistically significant difference in model performances. We therefore conduct Wilcoxon signed-rank tests with Bonferroni correction to identify significant pairwise differences between models. LR with learned features performed significantly better than NB with learned features ($p = 0.03$); SVM with predefined features outperforms NB with learned features ($p = 0.03$); LR with predefined features outperforms NB with learned features ($p = 0.03$); SVM with predefined features performs significantly better than NB with predefined features ($p = 0.03$); LR with predefined features outperforms NB with predefined features ($p = 0.03$). The results are displayed in Figure 5 in the Appendix B.

	Precision	Recall	F1-Score
SVM_TFIDF	0.83 (0.03)	0.81 (0.03)	0.81 (0.03)
NB_TFIDF	0.80 (0.03)	0.79 (0.03)	0.79 (0.03)
LR_TFIDF	0.82 (0.03)	0.82 (0.03)	0.82 (0.03)
SVM_LIWC	**0.83 (0.03)**	**0.83 (0.03)**	**0.83 (0.03)**
NB_LIWC	0.76 (0.4)	0.75 (0.04)	0.75 (0.04)
LR_LIWC	0.83 (0.03)	0.83 (0.03)	0.82 (0.03)

Table 1: Evaluation Metrics (Mean and STD)

6 Feature Importance

We employ different feature importance methods tailored to each model's learning mechanism to ensure that feature rankings are meaningful and aligned with the way each algorithm processes data. For the SVM models, we choose Linear SVM Coefficients because they directly represent feature importance in the decision boundary and are computationally efficient to extract. This method is fast and directly interpretable without requiring additional computations, but it does not capture feature interactions

or non-linearity. For the NB models, we choose Permutation Importance. NB does not have meaningful coefficients, and this method provides a model-agnostic way to assess how each feature affects predictions. This method allows the interpretation of feature contributions without relying on the model's internal parameters, but it is computationally expensive and can be sensitive to correlated features. For the LR models, we choose SHAP (SHapley Additive exPlanations [12]) Values, because they provide both global and instance-level feature attributions while considering feature interactions, making them more informative than raw coefficients. SHAP accounts for feature dependencies and offers a nuanced interpretation of how features contribute to individual predictions, but its computations can be slow and the results depend on the reference distribution used. Using SHAP for the SVM would be unnecessary because it would give similar results as the coefficients but less directly and with added computational cost, while SHAP's dependency assumptions conflict with NB's independence assumption. The contribution of each feature to the classification decision is indicated with a feature importance score. These are computed differently depending on the method: in Linear SVM Coefficients, they are derived from the absolute magnitude of the learned weights; in Permutation Importance, they are measured by assessing the decrease in model performance when a feature's values are randomly shuffled; while in SHAP, they quantify the contribution of each feature to the predicted classification probability by distributing the model's output among the input features.

6.1 SVM: Linear SVM Coefficients

For SVM, feature importance is determined using Linear SVM Coefficients. This method is chosen because linear SVM explicitly learns a set of coefficients as part of its optimisation process, making feature importance inherently interpretable. Additionally, since the SVM model is optimised to find the maximum margin, features with the largest coefficients contribute the most to defining this separation, allowing for a clear ranking of feature relevance. The resulting importance scores are based on the absolute magnitude of the learned coefficients, and like them, they can be any real value. While the importance scores' scale depends on the range of the input features, higher numbers indicate a stronger influence on classification. The top-3 features for the SVM models are *family, we*, and *with* (TF-IDF) and *social referents, I*, and *personal pronouns* (LIWC) (RQ3).

6.2 Naïve Bayes: Permutation Importance

For NB, we choose Permutation Importance because it provides a robust way to assess feature significance in probabilistic models that do not generate explicit importance scores. By quantifying the dependence of the model's predictions on each feature, Permutation Importance allows for an intuitive understanding of which features are most influential in the NB classification process. The scores produced are relative, and their scale depends on the model's performance metric; a larger value indicates that the feature has a greater impact on classification accuracy. The top-3 features for the NB models are *us, birthday*, and *her* (TF-IDF) and *social referents, social*, and *she/he* (LIWC) (RQ3).

6.3 Logistic Regression: SHAP Values

LR calculates the probability of a given outcome using a linear combination of input features, but SHAP offers a more granular and interpretable way of explaining these predictions. This method is chosen because it provides a comprehensive, intuitive, and theoretically grounded measure of feature importance, making it well-suited for interpreting the decision-making process of a probabilistic model like LR. In this study, we reduce the SHAP computation sample size from 50 to 20 to improve efficiency while maintaining representative feature importance insights. SHAP scores are measured in the same scale as the model's output and sum to the difference between the model's output and the expected output across all features. They can be positive (probability of classification increased) or negative (probability of classification decreased). Their magnitude reflects the strength of the feature's influence on the classification decision. The top-3 features for the SVM models are *with, we*, and *my* (TF-IDF) and *social referents, Social*, and *I* (LIWC) (RQ3).

6.4 Overall feature importance

To determine the top-20 most important features across all models trained on learned features and across all models trained on predefined features, we aggregate the feature importance scores from each model and sum them across all models. This is done to show which features are consistently influential regardless of the model; however, due to differences in how each method computes importance, the aggregated scores should be viewed as indicative rather than absolute measures of feature relevance. The top-10 features for the models trained on learned features are displayed in Figure 1, while those for the models trained on predefined features in Figure 2 (RQ3). Additionally, we identify unique features for each model, defined as those that appear in the top-10 for a specific model but not in others. Following, we report those referring to models trained on learned features.

- **SVM**: *my, team, she, our, he, we, with, friend, with my, their.*
- **Naïve Bayes**: *team, they are, he was, us, birthday, she was, of our, with her, person, spending time.*
- **Logistic Regression**: *my, she, our, and, good, he, my family, we, it, sleep.*

Following, we report those referring to models trained on predefined features.

- **SVM**: *sexual, Dic, Social, socrefs, feeling, we, Affect, Drives, insight, WC.*
- **Naïve Bayes**: *Dic, Social, socrefs, number, moral, feeling, we, focuspast, Drives, illness.*
- **Logistic Regression**: *Dic, Social, socrefs, pronoun, Analytic, feeling, we, Affect, focuspast, Drives.*

This helps us shed light on how different algorithms interpret the data; some overlap in the reported features occurs because the different algorithms, despite using distinct mechanisms to estimate importance, converge on similar cues that are consistently predictive of SS. We calculate the correlation between feature importance rankings across the different models by computing the Pearson correlation coefficient between the feature importance scores of each pair of models, using their respective importance values across all features. This is displayed in Figures 6 and 7 in the Appendix C. A high positive correlation indicates similar feature rankings and vice versa. The highest correlation is measured between SVM and LR models, while the lowest between NB and LR for models trained on learned features, and between SVM and NB for models trained on predefined features.

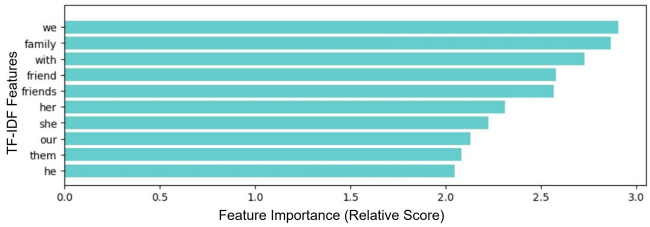

Figure 1: Top-10 Features for TF-IDF Models

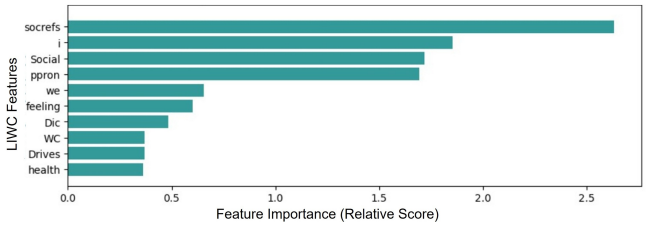

Figure 2: Top-10 Features for LIWC Models

7 Discussion

Our results indicate that the models trained on predefined features (LIWC) generally outperform those trained on learned features (TF-IDF n-grams), with the SVM model achieving the highest classification performance (RQ1-2). This suggests that LIWC features, which encapsulate linguistic and psychological constructs, provide a structured and interpretable representation of textual patterns related to SS. In contrast, TF-IDF captures surface-level word frequency distributions, which may be more susceptible to noise and context variability, limiting its predictive power for capturing abstract constructs like SS. Furthermore, our results support the findings by Caporusso et al. [4] regarding LIWC features correlated with SS. Notably, models trained on TF-IDF features tend to exhibit higher aggregated feature importance scores compared to those trained on LIWC. This could be attributed to the fact that TF-IDF operates on a larger and more granular feature space, capturing subtle variations in word usage. As a result, many features contribute partially to model decisions, leading to a higher sum of importance values across all features. In contrast, LIWC features are more constrained and predefined, leading to more concentrated but lower cumulative importance scores. This suggests that while TF-IDF captures a broader spectrum of textual variations, LIWC provides a more targeted and structured linguistic representation. Many of the features identified as relevant for the classification of SS (e.g., *we* and *social referents*) intuitively align with the nature of SS (RQ3).

8 Limitations and Future Work

This study serves as a pilot for the interpretable classification of different Self aspects in text, focusing on SS. Several areas for improvement remain. Clearer annotation guidelines are needed for consistency. The choice of restricting to linear models, LIWC features, and unigrams/bigrams was appropriate for this exploratory study prioritising interpretability; however, it inevitably limits performance and representational richness. In future work, we plan to complement this approach with more powerful models and richer feature sets (e.g., embeddings). Here we wanted to compare models trained on learned vs predefined features, but we plan to train models on both. While in this study we did not perform hyperparameter optimisation, we will do so in the future. We aim to train a neural network for multi-class classification, enabling simultaneous prediction of SS and other Self-aspects, allowing for a more comprehensive analysis of self-representation in text. In the future, we plan to employ different datasets and implement Demšar's evaluation method [8]. Our long-term goal is to be able, given a text instance, to determine what Self aspects are present and how they are expressed, in an explainable manner. To do so, it is not only necessary to extend our work to other Self-aspects, but to move beyond a binary classification for each of them. Work on the ontology underpinning future studies is ongoing [13].

9 Acknowledgments

We acknowledge Špela Rot's assistance and the financial support from the Slovenian Research Agency for research core funding for the programme Knowledge Technologies (No. P2-0103) and from the projects CroDeCo (J6-60109), Shapes of Shame in Slovene Literature (J6-60113), and Natural Language Processing for Corpus Analysis in the Medical Humanities (BI-VB/25-27-021). JC is a recipient of the Young Researcher Grant PR-13409.

References

[1] Ryan L Boyd, Ashwini Ashokkumar, Sarah Seraj, and James W Pennebaker. 2022. The development and psychometric properties of liwc-22. *Austin, TX: University of Texas at Austin*, 10.

[2] Marilynn B Brewer. 2002. Individual self, relational self, and collective self: partners, opponents, or strangers. (2002).

[3] Jaya Caporusso. 2022. Dissolution experiences and the experience of the self: an empirical phenomenological investigation (master's thesis). university of vienna. *Advisor: Assist. Prof. Dr. Maja Smrdu*.

[4] Jaya Caporusso, Boshko Koloski, Maša Rebernik, Senja Pollak, and Matthew Purver. 2024. A phenomenologically-inspired computational analysis of self-categories in text. In *Proceedings of JADT 2024*. Vol. 1, 169–178.

[5] Jaya Caporusso, Matthew Purver, and Senja Pollak. 2025. A computational framework to identify self-aspects in text. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*. Jin Zhao, Mingyang Wang, and Zhu Liu, editors. Association for Computational Linguistics, Vienna, Austria, (July 2025), 725–739. ISBN: 979-8-89176-254-1. DOI: 10.18653/v1/2025.acl-srw.47.

[6] Jaya Caporusso, Thi Hong Hanh Tran, and Senja Pollak. 2023. Ijs@ lt-edi: ensemble approaches to detect signs of depression from social media text. In *Proceedings of the Third Workshop on Language Technology for Equality, Diversity and Inclusion*, 172–178.

[7] Christopher G Davey and Ben J Harrison. 2022. The self on its axis: a framework for understanding depression. *Translational Psychiatry*, 12, 1, 23.

[8] Janez Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, 7, 1–30.

[9] Lewis R Goldberg. 2013. An alternative "description of personality": the big-five factor structure. In *Personality and Personality Disorders*. Routledge, 34–47.

[10] Boshko Koloski, Nada Lavrač, Bojan Cestnik, Senja Pollak, Blaž Škrlj, and Andrej Kastrin. 2024. Aham: adapt, help, ask, model harvesting llms for literature mining. In *International Symposium on Intelligent Data Analysis*. Springer, 254–265.

[11] X Alice Li and Devi Parikh. 2019. Lemotif: an affective visual journal using deep neural networks. *arXiv preprint arXiv:1903.07766*.

[12] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

[13] Luka Oprešnik, Tia Križan, and Jaya Caporusso. 2025. Building an ontology of the self: sense of agency and bodily self. In *Proceedings of Information Society 2025*. Cognitive Science. DOI: 10.70314/is.2025.cogni.8.

[14] James W Pennebaker, Matthias R Mehl, and Kate G Niederhoffer. 2003. Psychological aspects of natural language use: our words, our selves. *Annual review of psychology*, 54, 1, 547–577.

[15] Stephanie Rude, Eva-Maria Gortner, and James Pennebaker. 2004. Language use of depressed and depression-vulnerable college students. *Cognition & Emotion*, 18, 8, 1121–1133.

[16] Gemma Team et al. 2024. Gemma 2: improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

[17] David HV Vogel, Mathis Jording, Peter H Weiss, and Kai Vogeley. 2024. Temporal binding and sense of agency in major depression. *Frontiers in psychiatry*, 15, 1288674.

[18] Dan Zahavi. 2007. Self and other: the limits of narrative understanding. *Royal Institute of Philosophy Supplements*, 60, 179–202.

A Instructions for Labelling: Social Self

In the column relative to Social Self, insert:

- **0**: if the Social Self is not present.
- **1**: if the Social Self is present.

Following, we provide a definition of Social Self [4], instructions, and examples of a text instance where it is present and a text instance where it is not present, taken from the dataset to be labelled:

Definition: The Self as it is shaped and/or perceived when in an interaction or relationship of sorts with other people or entities to whom we attribute qualities of an inner life.

Instructions

For *Social Self* to be present in a text instance it is not enough for the text instance to contain references to other people and/or entities, but it has to contain mentions of the author's interactions with them, influence on them, or influence they have on the author. This can be even minimal, e.g., in the form of referring to a person as *my sister*, or by using the first-person plural pronoun instead of the singular one.

Examples

A.0.1 Text instance containing Social Self: "My family was the most salient part of my day, since most days the care of my 2 children occupies the majority of my time. They are 2 years old and 7 months and I love them, but they also require so much attention that my anxiety is higher than ever. I am often overwhelmed by the care they require, but at the same, I am so excited to see them hit developmental and social milestones."

Explanation of text instance with Social Self present: In this text instance, the author report on other people they are in some sort of relationship with, and about some aspects of their relationship and how they make the author feel.

A.0.2 Text instance not containing Social Self: "Yoga keeps me focused. I am able to take some time for me and breathe and work my body. This is important because it sets up my mood for the whole day."

Explanation of text instance with Social Self not present: In this text instance, the author does not report on any person, animal, or other entities to whom we attribute qualities of inner life.

General Notes While a certain Self-aspect might not be prominently present in a text instance in its entirety, if it is present in a part of the text instance to be labelled, then it has to be labelled as present in the text instance. A given text instance can have none of the Self-aspects present, one of them present and two of them non-present, two present and one non-present, or all three of them present—any combination is possible.

B Evaluation

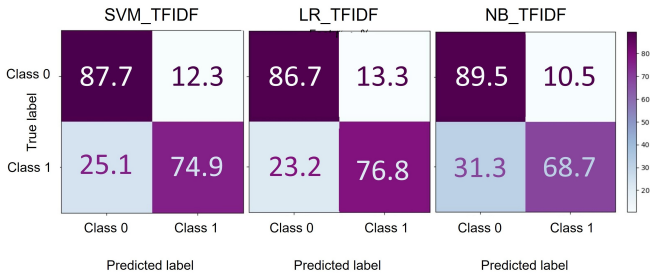

Figure 3: Confusion Matrices: Models Trained on Learned Features (TF-IDF)

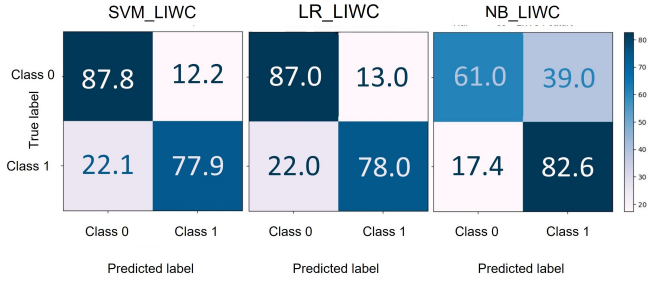

Figure 4: Confusion Matrices: Models Trained on Predefined Features (LIWC)

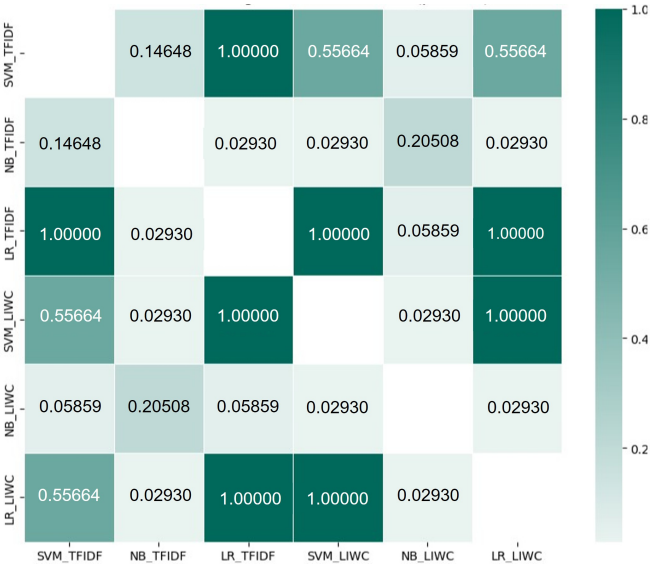

Figure 5: Pairwise Wilcoxon Signed-Rank Test Results (p-values)

C Feature Importance

Figure 6: Correlation Between Feature Importance Across Models Trained on Learned Features (TF-IDF)

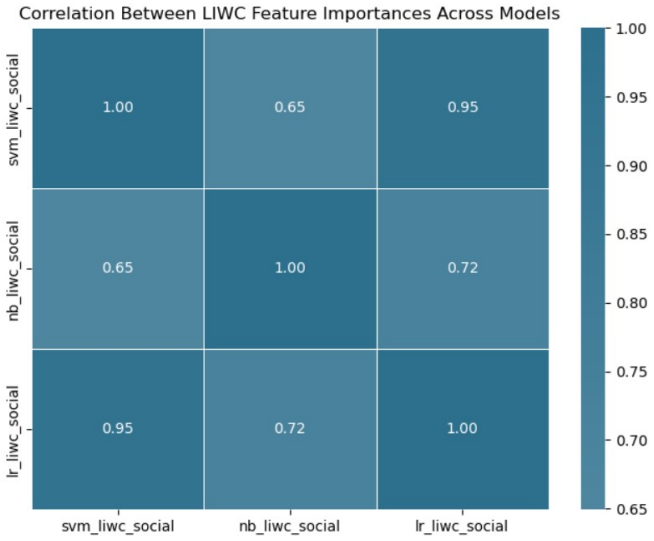

Figure 7: Correlation Between Feature Importance Across Models Trained on Pre-Defined Features (LIWC)