DUTH at SemEval-2023 Task 9: An Ensemble Approach for Twitter Intimacy Analysis

Georgios Arampatzis Vasileios Perifanis Symeon Symeonidis Avi Arampatzis

Database & Information Retrieval research unit,
Department of Electrical & Computer Engineering,
Democritus University of Thrace
{geoaramp,vperifan,ssymeoni,avi}@ee.duth.gr

Abstract

This work presents the approach developed by the DUTH team for participating in the SemEval-2023 Task 9: Multilingual Tweet Intimacy Analysis. Our results show that preprocessing techniques do not affect the learning performance for the task of multilingual intimacy analysis. In addition, we show that fine-tuning a transformer-based model does not provide advantages over using the pre-trained model to generate text embeddings and using the resulting representations to train simpler and more efficient models such as Multilayer Perceptron (MLP). Finally, we utilize an ensemble of classifiers, including three MLPs with different architectures and a CatBoost model, to improve the regression accuracy.

1 Introduction

Intimacy analysis is concerned with identifying and analyzing the level of emotional closeness between individuals using natural language (Pei and Jurgens, 2020). This form of analysis has a variety of applications in domains such as social media, healthcare and marketing, providing valuable insights into user behaviour, sentiment and preferences (Prager, 2005).

Despite the significance of intimacy as a reflection of social norms (Pei and Jurgens, 2020; Prager, 2005; Pei et al., 2022), there is limited availability of resources for analyzing it using machine learning techniques. To address this issue, Pei et al. (2022) processed and prepared a multilingual dataset consisting of tweets in ten languages, including English, Spanish, French, Portuguese, Italian, Chinese, Hindi, Korean, Dutch, and Arabic. This dataset has been made publicly available to evaluate the performance of computational tools that analyse intimacy.

This paper presents an ensemble intimacy regression scheme based on an extensive experimental study of several regressor models (Breiman, 2001; Chen and Guestrin, 2016; Prokhorenkova et al., 2018; Haykin, 1994) and text-to-feature representations (Ramos et al., 2003; Le and Mikolov, 2014; Conneau et al., 2020; Barbieri et al., 2022). Our study shows that training simple models using text embeddings generated by transformer-based models leads to effective and rapid training and accurate results. Our models achieve state-of-the-art performance without fine-tuning a huge and training-costly transformer-based model. We carefully optimized the components of our final ensemble, considering the influence of the imbalanced regression task. Our team achieved a rank of 7 in the overall score, 25 using the languages seen during training, and 2 on cross-language transfer learning.

The remainder of this work is structured as follows. The proposed system is described in Section 2. Section 3 reports on our experiments. Finally, conclusions and future directions are discussed in Section 4.

2 System Description

The primary objective of SemEval-2023 Task 9 was to assign a continuous value within the range of $[1, 5]$ to multilingual tweet texts, representing their intimacy level. The higher the value, the higher the intimacy level. This section presents a detailed description of our system, elucidating the text-to-feature generation methods used. In addition, we provide statistics about the given dataset.

2.1 Dataset

The task organisers released the complete data in the training and testing sets. The training set comprised 9,491 texts, each annotated with a continuous value reflecting the level of intimacy. The test set consisted of 13,697 texts from ten different languages mentioned in the Introduction. The training set consisted of six languages: English, Spanish, Portuguese, Italian, French, and Chinese. Table 1 illustrates the number of samples per language on

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1225–1230

Language	#Train	#Test
English	1587	1396
Spanish	1592	1396
Portuguese	1596	1390
Italian	1532	1352
French	1588	1382
Chinese	1596	1354
Hindi	-	1260
Dutch	-	1389
Korean	-	1410
Arabic	-	1368

Table 1: Number of samples per language

both the training and testing sets. Notably, the training and testing sets exhibit balanced examples with respect to language. However, we observed a high level of skewness in the intimacy level per language, resulting in an imbalanced regression task. Figure 1 presents the distribution of the intimacy level on the training set per language. Our analysis shows that, in all cases, the distribution of texts per language is skewed, indicating a challenging learning task.

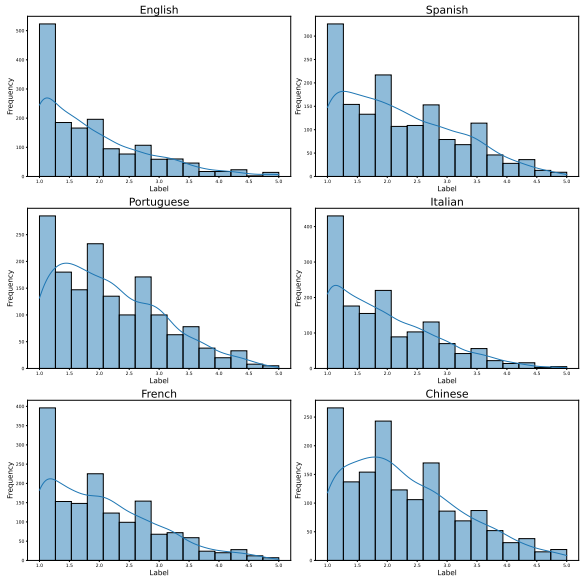

Figure 1: Distribution of intimacy level per language.

The majority of observations exhibit a low intimacy score, implying that most tweets are not intimate. To mitigate the effect of the skewed distribution, we employed the LogCosh loss function to update a model's weights whenever applicable. We selected LogCosh over conventional criteria such as MSE and MAE, empirically and based on its robustness to outliers.

2.2 Pre-Processing

We follow the techniques presented in (Symeonidis et al., 2018) for pre-processing. We convert all tweets to lowercase and remove multiple whitespaces. Then, we examined the impact of various pre-processing techniques on regression accuracy. These techniques include removing HTML tags, converting numbers and mathematical symbols to words, replacing `@user` with `atUser`, removing Unicode characters by language, replacing repetitions, contractions, elongated words, URL addresses, hashtags and stopwords, as well as the application of stemming and/or lemmatization.

Our experimental results, as we will see later, indicate that pre-processing techniques have minimal impact on the regression accuracy for multilingual tweet intimacy analysis.

2.3 Text-to-Feature Representation

For representing tweets as feature vectors, we consider the following methods:

- CountVectorizer is a widely used method for converting text documents into numerical features based on token counts. This method ignores the order of words in the document, focusing only on their frequency.

- TF-IDF is another widely used technique for document representation. It measures the importance of a term in a document by assigning a higher weight to words that appear more frequently in the document, improving the overall representation.

- Doc2Vec (Le and Mikolov, 2014) is a neural network model that learns a vector representation of documents. It captures the meaning of words within a document, providing a more accurate analysis at the document level than CountVectorizer and TF-IDF.

- XLM-R (Conneau et al., 2020) is a pre-trained transformer that models multilingual representations. It is designed for cross-lingual language understanding and achieves state-of-the-art performance in various multilingual tasks.

- XLM-T (Barbieri et al., 2022), which is based on XLM-R, is a transformer model designed explicitly for multilingual tweet-based

datasets. XLM-T has been shown to outperform other state-of-the-art approaches in tasks such as sentiment analysis.

Our experimental study, as we will see later, indicates that training on transformer-based embeddings results in significant improvements with respect to regression accuracy compared to traditional methods.

2.4 Machine Learning Models

The system implementation uses Python 3.10 and relies on several open-source frameworks, including scikit-learn, PyTorch, CatBoost, XGBoost and transformers. The following regression models were employed:

- Linear Regression is a simple baseline that models the relationship between a dependent variable and one or more independent variables.
- k-Nearest Neighbors is a lazy learning method, identifying the k training instances closest to a given sample.
- Random Forest (Breiman, 2001) is an ensemble approach that combines multiple decision trees to predict the value of a given sample.
- XGBoost (Chen and Guestrin, 2016) is a popular gradient-boosting algorithm by iteratively training a sequence of weak decision trees and adjusting the weights for the wrong predicted instances.
- CatBoost (Prokhorenkova et al., 2018) is another gradient-boosting algorithm using a combination of ordered boosting, random permutations and gradient-based sampling.
- MLP (Haykin, 1994) is a simple feedforward neural network consisting of multiple layers of interconnected nodes. Unless stated otherwise, the architecture of the MLP consists of one hidden layer with 64 units.

Our experimental results, as we will present after, indicate that both the CatBoost and MLP regressors attain state-of-the-art performance in multilingual tweet intimacy analysis. Notably, the performance achieved by these models is comparable to the fine-tuning of large transformer-based models.

2.5 Evaluation Measures

The evaluation measure the task organisers consider is the Pearson correlation coefficient (r) between the predicted and actual sentiment levels. Additionally, given that we deal with a regression task, the Mean Squared Error (MSE) and Mean Absolute Error (MAE) were calculated as commonly used measures for evaluating model performance.

3 Experiments

This section presents the results from utilizing the pre-processing techniques, vectorizers, and machine learning models discussed in Section 2. Additionally, we compare our findings with the corresponding reported results obtained from fine-tuned transformers. Unless stated otherwise, all experiments used 5-fold cross-validation after shuffling the dataset to improve generalisation.

3.1 Impact of Pre-processing Techniques

We begin our experimental study by assessing the influence of pre-processing on regression accuracy. We select CatBoost as the learning algorithm, a model known for its fast training and high predictive accuracy in related tasks, and the CountVectorizer which enables rapid text conversion to feature representations.

Model	MSE	MAE	r
Raw text	0.6985 ± 0.0390	0.6645 ± 0.0095	0.3873 ± 0.0261
HTML tags removal	0.6988 ± 0.0365	0.6646 ± 0.0125	0.3872 ± 0.0222
Number to word	0.6955 ± 0.0374	0.6624 ± 0.0136	0.3919 ± 0.0239
Math symbol to word	0.6946 ± 0.0388	0.6621 ± 0.0141	0.3935 ± 0.0245
user replacement	0.6947 ± 0.0365	0.6628 ± 0.0129	0.3931 ± 0.0216
Unicode removal	0.6841 ± 0.0371	0.6575 ± 0.0144	0.4094 ± 0.0187
Repetition replacement	0.6985 ± 0.0390	0.6645 ± 0.0140	0.3873 ± 0.0261
Contractions replacement	0.6966 ± 0.0395	0.6632 ± 0.0144	0.3900 ± 0.0261
Elongated replacement	0.6975 ± 0.0389	0.6637 ± 0.0137	0.3890 ± 0.0252
URL replacement	0.6980 ± 0.0373	0.6641 ± 0.0130	0.3884 ± 0.0235
Stopwrods removal	0.7493 ± 0.0484	0.6921 ± 0.0159	0.2981 ± 0.0258
Stemming	0.6939 ± 0.0376	0.6621 ± 0.0125	0.3953 ± 0.0225
Lemmatization	0.6993 ± 0.0400	0.6650 ± 0.0139	0.3861 ± 0.0218

Table 2: Evaluation of pre-processing techniques

From Table 2, it is evident that pre-processing techniques have minimal impact on learning performance. While some techniques, such as numbers to words and Unicode removal, have demonstrated a slightly higher Pearson's r correlation coefficient, the difference amounts to only approximately 2%. This observation holds by applying combinations of pre-processing techniques, which are not included in the manuscript due to space limitations. Thus, we assert that pre-processing techniques have little significance in the context of intimacy analysis, in contrary to the previous

study of Symeonidis et al. (2018). Consequently, we proceeded with experiments using raw text data.

3.2 Encoders and Machine Learning Model Evaluation

We continue our experimental study using the raw data and 5-fold cross-validation without applying pre-processing techniques. We utilize the vectorizers along with the machine learning models mentioned in Section 2. We have chosen not to fine-tune a transformer-based model due to their time-consuming training.

Figure 2 illustrates the average Pearson's r for each considered model and vectorizer. It is evident that transformer-based embeddings, specifically XLM-R and XLM-T, exhibit a significant improvement compared to CountVectorizer, TF-IDF, and Doc2Vec. It is worth noting that Doc2Vec fails to achieve convergence, which we attribute to the nature of our multilingual task.

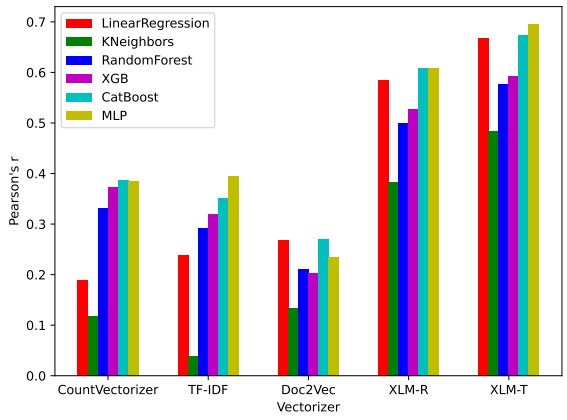

Figure 2: Average Pearson's r per model and vectorizer

To calculate the relative improvement (RI), we consider the MLP model and use the following formula:

$$\mathrm{RI} = ((r_{\mathrm{new}} - r_{\mathrm{old}})/r_{\mathrm{old}}) \times 100, \qquad (1)$$

where r_{new} denotes the Pearson's r for the transformer-based embeddings and r_{old} is the Pearson's r for the TF-IDF vectorizer. The results show a remarkable improvement of approximately 54.5% and 76.5% for XLM-R and XLM-T, respectively, compared to TF-IDF in terms of performance gain. Comparing the relative improvement between XLM-R and XLM-T, the latter achieves approximately 14.2% higher r.

Regarding model-specific results, we observe that MLP and CatBoost outperform the other models in all considered cases.

For completeness, Table 3 presents the MSE and MAE obtained for each model and vectorizer to present sufficiently the Regression experimental results. The highest-performing model per vectorizer is denoted in bold, and the second-best model is underlined. The results confirm that MLP and CatBoost outperform the other models and that the XLM-T model performs better than the XLM-R model.

Vectorizer	Model	MSE	MAE
CountVectorizer	LinearRegression	2.4091±0.2840	1.1610±0.0611
	KNeighbors	1.1869±0.1038	0.8423±0.0359
	RandomForest	0.8195±0.0551	0.6829±0.0196
	XGB	0.7026±0.0295	0.6744±0.0101
	CatBoost	**0.6988**±0.0365	**0.6646**±0.0125
	MLP	0.7148±0.0325	0.6653±0.0210
TfidfVectorizer	LinearRegression	1.3508±0.1112	0.9021±0.0383
	KNeighbors	0.9893±0.0881	0.8144±0.0586
	RandomForest	0.8274±0.0577	0.6992±0.0214
	XGB	0.7414±0.0346	0.6907±0.0103
	CatBoost	0.7221±0.0434	0.6740±0.0144
	MLP	**0.6959**±0.0359	**0.6663**±0.013
Doc2Vec	LinearRegression	**0.7597**±0.0404	0.7053±0.0154
	KNeighbors	0.9974±0.0391	0.7934±0.0145
	RandomForest	0.8050±0.0327	0.7274±0.0091
	XGB	0.8536±0.0374	0.7394±0.0111
	CatBoost	0.7619±0.0431	**0.6959**±0.0122
	MLP	0.7783±0.0481	0.7034±0.0151
XLM-R	LinearRegression	0.5447±0.0180	0.5845±0.0071
	KNeighbors	0.8312±0.0191	0.7295±0.0081
	RandomForest	0.6155±0.0228	0.6350±0.0084
	XGB	0.6038±0.0021	0.6171±0.0440
	CatBoost	**0.5153**±0.0231	0.5681±0.0091
	MLP	0.5160±0.0191	**0.5606**±0.0087
XLM-T	LinearRegression	0.4582±0.0183	0.5304±0.0061
	KNeighbors	0.6693±0.0392	0.6366±0.0176
	RandomForest	0.5491±0.0219	0.5923±0.0201
	XGB	0.5404±0.0212	0.5748±0.0118
	CatBoost	0.4472±0.0183	0.5241±0.0043
	MLP	**0.4215**±0.0157	**0.5036**±0.0063

Table 3: MSE and MAE per model and vectorizer

3.3 Cross-language Transfer Learning

Based on the findings in the previous section, we have opted to use MLP and CatBoost for evaluating the cross-language transfer learning task using XLM-T. For this task, we have randomly selected a subset of the given dataset to act as the validation set. Moreover, we exclude the text from a single language it has never seen during training. With the cross-language task, we ensure the generalization and robustness of the considered approach.

Figure 3 presents the obtained Pearson's r by excluded language. Overall, both MLP and CatBoost perform similarly across all languages, with MLP outperforming CatBoost in all cases. Interestingly, both classifiers achieve their highest performance score in English, which may be due to the large English corpus of the pre-trained model. The lowest

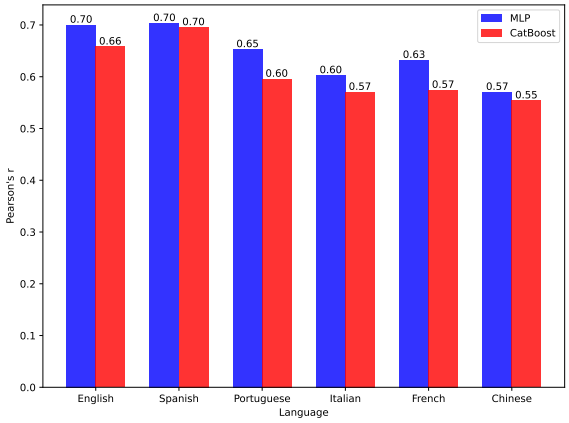

Figure 3: Pearson's r for the cross-language task

accuracy scores are obtained for Chinese, which is a more challenging language for natural language processing due to its complex character set and grammar.

3.4 Final Ensemble Model and Results

In the previous experimental sections, we demonstrated that combining XLM-T with MLP or CatBoost outperforms other approaches. In our final solution, we employ an ensemble consisting of three MLPs and the CatBoost model. MLP1, which was considered for all previous experiments, has a single hidden layer with 64 units. MLP2 has three hidden layers of size 256, 168, and 64. MLP3 also has three hidden layers of size 1024, 128, and 64.

All models are trained by optimizing the Log-Cosh loss function due to the imbalanced nature of the task, as mentioned in Section 2. The dataset is randomly split into 80% for training and 20% for testing. After training, the results indicate that MLP3 $>$ MLP2 $>$ MLP1 $>$ CatBoost in terms of Pearson's r. For the final solution, a weighted voting approach is utilized using the formula

$$\text{Prediction} = 0.1\,\text{MLP1} + 0.3\,\text{MLP2} + 0.5\,\text{MLP3} + 0.1\,\text{CatBoost} \quad (2)$$

which is found based on a small grid search, which involves tuning the weights assigned to each model and ensuring that their sum equals one. The selection of optimal weighting is based on the performance of the ensemble method measured by the highest r observed in the test set.

The results of the organizer's test set across all languages, as well as the comparison with fine-tuned transformers (obtained from (Pei et al., 2022)), are presented in Table 4 for a Pearson's r. XLM-T and XLM-R appear to perform relatively well across all languages, with Hindi being the only language with significantly lower performance. Distill, MiniLM and BERT perform somewhat worse, with BERT showing poor performance across all languages except for English and Spanish. Interestingly, our model's performance, consisting of an ensemble of models trained with the embeddings generated by the pre-trained XLM-T, is consistently high across all languages, outperforming all other models except for XLM-T in some cases.

Language	XLM-T	BERT	XLM-R	Distill	MiniLM	Ours
English	0.70	0.59	0.65	0.55	0.61	0.70
Spanish	0.73	0.62	0.64	0.61	0.67	0.70
Portuguese	0.65	0.54	0.61	0.52	0.53	0.66
Italian	0.70	0.57	0.67	0.58	0.62	0.69
French	0.68	0.55	0.63	0.54	0.57	0.68
Chinese	0.70	0.65	0.72	0.67	0.65	0.69
Hindi	0.24	0.09	0.24	0.17	0.18	0.23
Dutch	0.59	0.47	0.60	0.44	0.57	0.63
Korean	0.35	0.32	0.33	0.26	0.41	0.33
Arabic	0.64	0.35	0.48	0.32	0.38	0.60
overall	0.58	0.48	0.53	0.52	0.53	0.60

Table 4: Final results: Performance of the baselines and the proposed approach concerning Pearson correlation r

4 Conclusion

Twitter provides a platform for users to express personal opinions and share emotions, making it an attractive research area for analyzing intimacy and social norms. This work presents an approach for Twitter sentiment analysis that involves training simple and efficient models using embeddings generated by pre-trained transformers. Our approach is comparable to fine-tuning large transformer models and proposes that the complex and challenging task of fine-tuning transformer models can be replaced with more straightforward and highly efficient models trained using embeddings derived from pre-trained transformers. This conclusion could potentially lead to significant improvements in the efficiency and effectiveness of natural language processing and machine learning applications. A limitation of the presented method for Intimacy analysis is that it does not address the issue of imbalanced regression tasks, which could compromise the quality of the results. Future work in this domain could appropriately address the imbalanced regression task to achieve higher-quality outcomes.

References

Francesco Barbieri, Luis Espinosa Anke, and José Camacho-Collados. 2022. XLM-T: multilingual language models in twitter for sentiment analysis and beyond. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June 2022*, pages 258–266. European Language Resources Association.

Leo Breiman. 2001. Random forests. *Mach. Learn.*, 45(1):5–32.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 785–794. ACM.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Simon Haykin. 1994. *Neural networks: a comprehensive foundation*. Prentice Hall PTR.

Quoc V. Le and Tomás Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1188–1196. JMLR.org.

Jiaxin Pei and David Jurgens. 2020. Quantifying intimacy in language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5307–5326, Online. Association for Computational Linguistics.

Jiaxin Pei, Vítor Silva, Maarten Bos, Yozon Liu, Leonardo Neves, David Jurgens, and Francesco Barbieri. 2022. Semeval 2023 task 9: Multilingual tweet intimacy analysis.

Karen J. Prager. 2005. *The psychology of intimacy*. Guildford Press, New York.

Liudmila Ostroumova Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6639–6649.

Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Symeon Symeonidis, Dimitrios Effrosynidis, and Avi Arampatzis. 2018. A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. *Expert Syst. Appl.*, 110:298–310.