Journal of Artificial Intelligence and Data Science (JAIDA)
e-ISSN: 2791-8335
https://jaida.ikcu.edu.tr

Vol. 1, No. 2, pp. 144-149, 2021
Research Article

Performance Trade-Off for Bert Based Multi-Domain Multilingual Chatbot Architectures

Davut Emre TAŞAR [1,*], Şükrü OZAN [2], Seçilay KUTAL [3], Oğuzhan ÖLMEZ [4], Semih GÜLÜM [5], Fatih AKCA [6], Ceren BELHAN [7]

[1] Dokuz Eylül University, Turkey, davutemre.tasar@deu.edu.tr
[2] AdresGezgini A.S¸., Ar-Ge Merkezi, Turkey, sukruozan@adresgezgini.com
[3] Areal AI, U.S., secilay.kutal@areal.ai
[4] Manisa Celal Bayar University, Turkey, 172803041@ogr.cbu.edu.tr
[5] AVL Araştırma ve Mühendislik, Turkey, Semih.Gulum@avl.com
[6] Sakarya University, Turkey, mehmet.akca1@ogr.sakarya.edu.tr
[7] Software Engineering, Izmir University of Economics, Turkey, ceren.belhan@std.ieu.edu.tr

Abstract

Text classification is a natural language processing (NLP) problem that aims to classify previously unseen texts. In this study, Bidirectional Encoder Representations for Transformers (BERT) architecture is preferred for text classification. The classification is aimed explicitly at a chatbot that can give automated responses to website visitors' queries. BERT is trained to reduce the need for RAM and storage by replacing multiple separate models for different chatbots on a server with a single model. Moreover, since a pre-trained multilingual BERT model is preferred, the system reduces the need for system resources. It handles multiple chatbots with multiple languages simultaneously. The model mainly determines a class for a given input text. The classes correspond to specific answers from a database, and the bot selects an answer and replies back. For multiple chatbots, a special masking operation is performed to select a response from within the corresponding bank answers of a chatbot. We tested the proposed model for 13 simultaneous classification problems on a data set of three different languages, Turkish, English, and German, with 333 classes. We reported the accuracies for individually trained models and the proposed model together with the savings in the system resources.

Keywords: BERT; classification; chatbot; memory gain; multi-domain; multi-lingual.

1. Introduction

As a sub-discipline of artificial intelligence (AI), Natural Language Processing (NLP) aims to solve text classification, sentiment analysis, and summary information extraction using text data. Today, developments in NLP are also gaining momentum. Live chatbots produced using natural language processing have now reached human-level performance.

In [1], a chatbot model based on Long Short-Term Memory (LSTM) has been created to automatically answer frequently asked questions specific to a related industry with 83% accuracy. In their study, Ozan et al. compared the performances of LSTM, BERT, and Doc2Vec on the categorical classification problem [2]. They concluded that the most successful model was the BERT model with 93% accuracy.

In a previous study, we investigated pre-trained transformer models specific to the Turkish language. We figured out that the most successful model was loodos/bertbase-turkish-cased with a test accuracy score of 89% [3]. This study proposes a system used in a live chatbot previously designed in [4] to select the most appropriate response to a request. The BERT model is preferred as the core language model. Considering the classical understanding that has continued until today, each model trained for live chatbot architectures needs a separate area in memory. We aim to introduce a single model with the proposed model that can simultaneously solve multiple text classification problems. We saw that reducing the solution to a single model can be an effective optimization method for RAM required by numerous simultaneous chatbots.

In [5], Tellez et al. studied the sentiment analysis problem for seven different languages. They used the data collected from Twitter. Baseline for Multilingual Sentiment Analysis model (b4msa) was used as a bootstrapping sentiment classifier by fine-tuning the test results. They achieved a 79.9% accuracy.

We created some of our training data by translating our original Turkish text data to other languages using machine translation as described in [7]. In [7], Xiaochuan et al. successfully classified entry titles of different

[*] Corresponding author e-mail address: davutemre.tasar@deu.edu.tr

Received: 30/11/2021; Accepted: 24/12/2021

languages in Wikipedia. Even though a given title is in a different language, their proposed system could classify them according to the title subject.

In [8], Schwenk et al. performed sentiment analysis of the data in RCV2 data set using morphologically and grammarly different eight different languages. They trained and tested different language models using a balanced data set, i.e., approximately equal number of data for each language.

Bilingual bi-directional LSTM model is preferred for cross-lingual sentiment classification problems in [8]. Zhou et al. performed sentiment analysis with four different models. They combined the LSTM model with both sentence-based, word-based, and sentence-word-based attention models. They achieved the highest accuracy of 82.4% with the latter model configuration.

2. Methodology

The transformer-based BERT model was preferred to solve the problem of reducing classification problems to a single model. For the developed chatbot to serve more than one company and provide its service in more than one language, data sets belonging to different languages were studied. It is ensured that the model can make company-specific estimations. Also, the performance loss due to a single model used in the structure is minimized using the estimation function.

2.1. Dataset

To increase the complexity, we used multiple data sets for each language. We collected the primary data set of English texts. Then we generated data set for other languages by translating the primary data set using machine translation. We guaranteed our BERT classifier model to learn semantics rather than the language itself by increasing the complexity. We used three languages, English, Turkish and German, and four data sets. Four separate data sets, which will be named as "dataset1 and dataset2", "dataset3", "dataset4" and "dataset5", were included. These data sets were divided among themselves, and the number of data sets was increased to 13.

Table 1. *Example sentences according to the subsets in the dataset and ratios of sub-datasets to the whole dataset (the three dots represent the continuation of the sentence).*

Dataset Name	Sample Text	Ratio of the subdataset to the whole dataset
Dataset1GER	wie kommt ich um mein geld aus dem geldautomaten zu bekommen	%2.88
Dataset2GER	wie kann ich mein konto löschen	%2.75
Dataset3GER	grant morrison, das mit göttern spricht, ist ein dokumentierender merkmalslänge der eine…	%11.07
Dataset4GER	das abingdon und witney college ist ein welteres bildungskollegium in abingdon oxfordshire...	%13.56
Dataset1EN	i want to start using my card how do i active it	%2.88
Dataset2EN	i want to block my card or deactivate it or something it s been stolen and i don t want it misused...	%2.75
Dataset3EN	count the number of food items on list	%11.07
Dataset4EN	the saul river is a tributary of the izvorul river in romania	%13.56
Dataset5EN	i m still feeling pretty low and demotivated including ups	%9.24
Dataset1TR	size birkaç gün önce bir çek gönderdim henüz hesabıma hiçbir şey olmadı bu kabul edilemez param nerede	%2.88
Dataset2TR	sanal kart nasıl alınır	%2.75
Dataset3TR	tom un telefon numarası nedir	%11.07
Dataset4TR	pointe magazine bale dansçılarına yönelik uluslararası bir dergidir ve macfadden sonra sahne sanatları medyası...	%13.56

The data set named "dataset1 and dataset2" has 77 classes and consists of short question sentences belonging to the banking sector and the categories to which the sentences belong [9]. This data set, taken from the Huggingface datasets library, was first translated into Turkish and German with the Translator library, using the

Microsoft Translation API in Python programming language. Afterward, the translation quality of the translated sentences was checked by the article's authors on 500 randomly selected samples, and incorrect translations were removed from the data set. This way, 13083 translated short speech sentences, and their categories were obtained as a usable data set. The dataset holds some similar questions for different use cases. Hence, this data set was divided into 38 and 39 classes according to the number of classes (dataset1 and dataset2) to test the BERT model's performance correctly. We used these sets as data sets belonging to different user groups were used in the model's training.

The "dataset3" was created by selecting 14 tag classes from DBpedia 2014 [10]. Although there were 630K rows of data in the original data set, we only used 31.500 rows, which ensured the data distribution was more balanced.

The data set named "dataset4" has 18 classes. It is categorized as a database of commands for human-robot interaction [12], containing 25.716 labeled commands. The data set named "dataset5" is a data set prepared for sentiment analysis and includes a total of six classes according to the emotional states of the texts in English [11].

It was noticed that there were too many wrong results in translating from Turkish to German. Hence, as shown in Figure 5, only dataset5 was left in its original language and was not translated into Turkish or German by machine translation. In addition, the regular distribution observed among the languages in the data set cannot be seen among the 5 data sets taken as a basis in the translation phase.

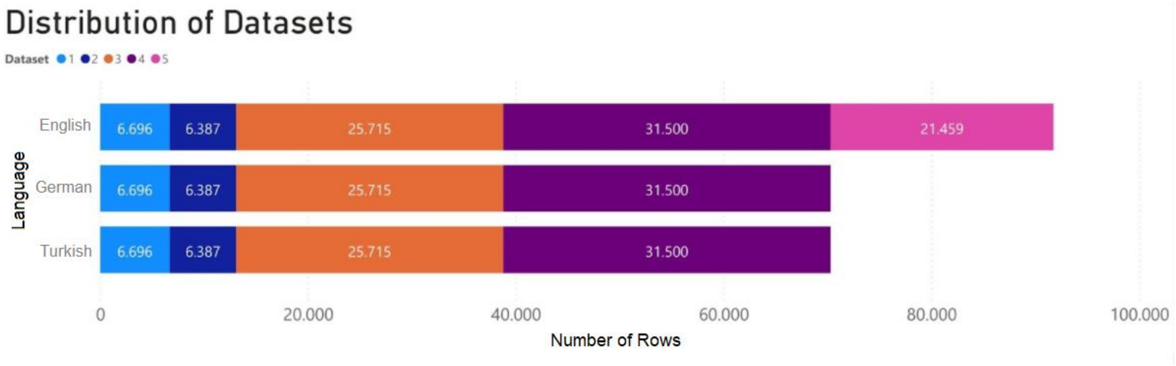


Figure 1. *The number of data contained in each dataset used in training and treated as a separate chatbot company.*

All sub-datasets are split at 70% train - 30% test data ratios. The data sets separated as trains were combined among themselves, and the data sets separated as tests were combined among themselves to form train and test data sets used in model training. In this way, in the model where all data sets are trained together, the distribution of the data sets is ensured to be the same as in the models trained separately for each data set.

2.2. BERT Method

BERT [13], a model developed by Devlin et al., was used to solve the classification problems in the article. The training of the BERT model, which is a bidirectional transformer model consisting of 12 transformer blocks, 12 self-attention heads, and 768 hidden layers, consists of two stages. In the first stage, the pre-training, which is carried out with high-dimensional data, the model is expected to learn the language structure. In the second stage, fine-tuning, the model is retrained with the data set based on the problem. In this process, the model creates a problem-specific attention mechanism and reinforces its predictions about the problem.

We used a pre-trained multilingual BERT model [13] in our study. It can serve more than one language within the chatbot architecture. We previously hit the highest performance in the categorical classification problem in [3] with the corresponding model. This model has been pre-trained with Wikipedia data on 104 languages. We fine-tuned the model with 13 datasets explained in Section 2.1. The models were trained for five epochs, and the training parameters were kept constant for each training.

2.3. Decision Function

Each model takes up approximately 2 GB of memory when transferred to memory. This memory requirement differs with respect to the model architecture. In cases where more than one model needs to be used, as in this study, this number is multiplied by the number of models used to calculate memory usage. For this reason, as the number of models used increases, the area used in memory also increases.

$$Memory\ Space\ Gain = x_{size}(\ n_{model} - 1) \tag{1}$$

The memory space gain provided by the method proposed in the study is shown in (Equation 1). Memory space gain is calculated by multiplying the number of models (n_{model}) minus one by the size of the fine-tuned BERT model (x_{size}).

Considering a structure that provides chatbot service to more than one user, it is possible to access the information about which problem came from which user. In the study, the classes that the created model will predict are limited with this information. The proposed single model approach finds the correct output class by applying softmax within a corresponding dataset's masked output.

3.Results

At the stage of evaluating the performance of the models whose training has been completed, the complexity matrix, which is formed by binary classification according to the number of correct and incorrect predictions in the test data, is taken as a basis. If the value predicted by the model is correct, the True value increases. If the true value and the predicted value are positive, the number of True Positive (TP) increases, and if both are negative, the number of True Negative (TN) increases. However, if the class predicted by the model is wrong, the value of False increases. While the true value is positive, the False Negative (FN) value increases when the estimated value is negative, while the False Positive (FP) value increases when the true value is negative and the predicted value is positive.

The performance of the models was evaluated considering the accuracy and F1 score metrics calculated over the complexity matrix. Accuracy (Equation 2) expresses what percentage of the predictions produced by the model are correct. F1 score (Equation 3), which is frequently preferred in the evaluation of multiple classification models, is a metric in which not only the correctness of the classification outputs but also the incorrect predictions of the model can be expressed. It is calculated over the harmonic mean of the model's precision (Equation 4) and recall metrics (Equation 5). Instead of calculating the F1-score over all the predictions of the model regardless of the classes of the problem, the averages of the F1-score values calculated for the classes are summed to obtain a macro F1-score (Equation 6). Thus, it can be ensured that each class is evaluated as a separate binary classification problem. In order to make the F1-score better able to express the class distribution, the weighted F1-score (Equation 7) can be calculated by taking into account the distribution ratios of the classes in the dataset.

$$Accuracy = \sum(TP + TN)/\sum(TP + TN + FP + FN) \tag{2}$$

where TP is True Positive count, TN is True Negative count, FP is False Positive count and FN is False Negative count.

$$F1 - score = (2 * Precision * Recall)/(Precision + Recall) \tag{3}$$

where Precision and Recall are defined by,

$$Precision = \sum TP\ /\ \sum(TP + FP) \tag{4}$$

$$Recall = \sum TP\ /\ \sum(TP + FN) \tag{5}$$

$$Macro\ F1 - score = \sum_0{}^{i} F1_i\ /i \tag{6}$$

$$Weighted\ F1 - score = \sum_0{}^{i} F1_i * DistributionRatio_i\ /i \tag{7}$$

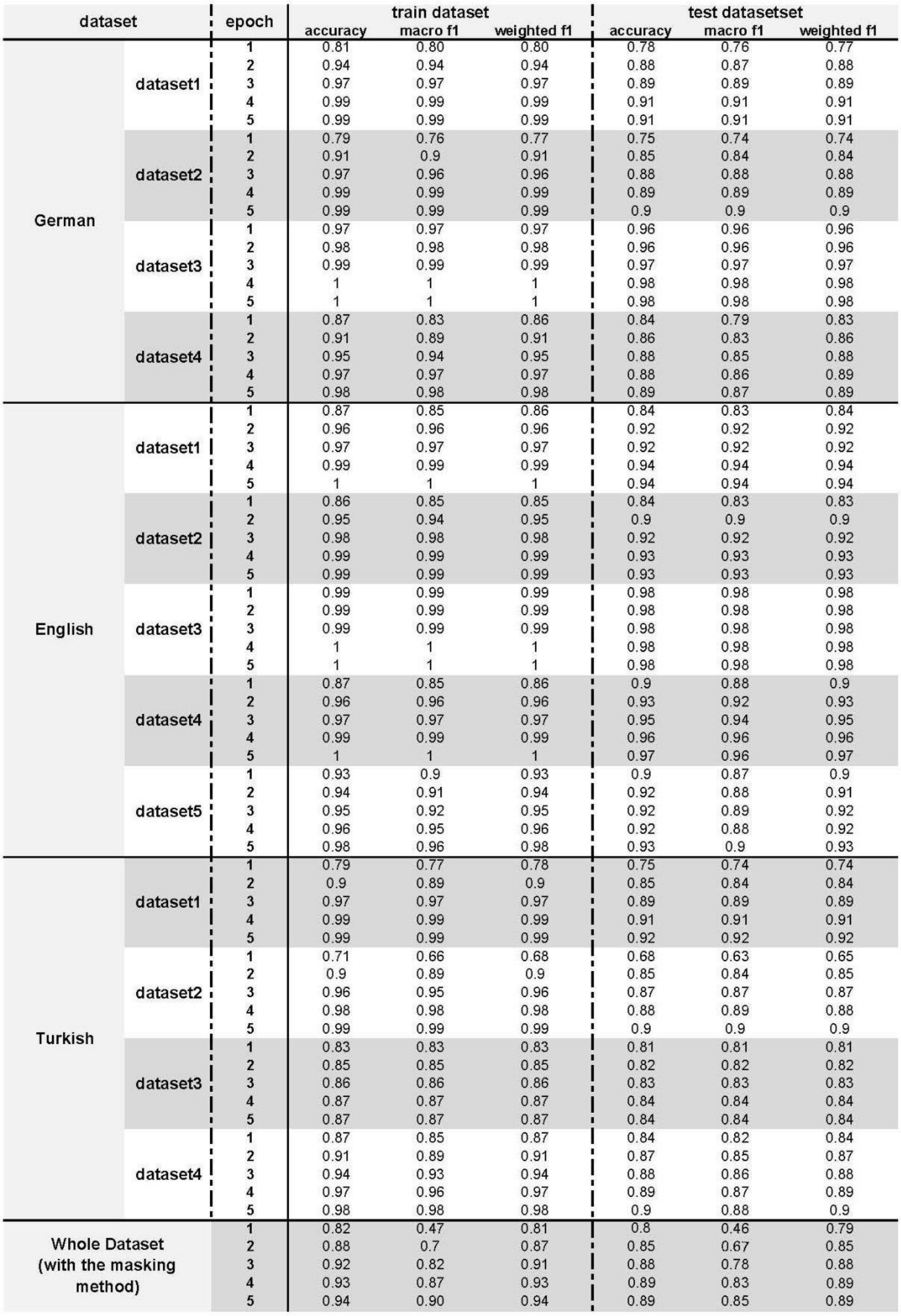
where i is the count of classes of the classification problem and F1 is F1-score.

14 models, which were trained in 5 terms, were tested on training and test data sets. The mentioned performance metrics were obtained for each period of the models and the results are given in Table 2.

During the estimation of the model, the contribution of the estimation function, which includes preliminary information about which company the data sets belong to, and thus provides the opportunity to make a prediction specific to the data set, to the multilingual model is also examined. The trained model was tested on a fixed test data set with and without using the estimation function. 80% accuracy was obtained with the test performed without utilizing the estimation function. The accuracy increased to 91% when the estimation function was used.

When both accuracy values were compared, it was concluded that the performance of the model increased significantly by the masking method.

Table 3. *Results of trained models by various metrics.*



dataset		epoch	train dataset			test datasetset		
			accuracy	macro f1	weighted f1	accuracy	macro f1	weighted f1
German	dataset1	1	0.81	0.80	0.80	0.78	0.76	0.77
		2	0.94	0.94	0.94	0.88	0.87	0.88
		3	0.97	0.97	0.97	0.89	0.89	0.89
		4	0.99	0.99	0.99	0.91	0.91	0.91
		5	0.99	0.99	0.99	0.91	0.91	0.91
	dataset2	1	0.79	0.76	0.77	0.75	0.74	0.74
		2	0.91	0.9	0.91	0.85	0.84	0.84
		3	0.97	0.96	0.96	0.88	0.88	0.88
		4	0.99	0.99	0.99	0.89	0.89	0.89
		5	0.99	0.99	0.99	0.9	0.9	0.9
	dataset3	1	0.97	0.97	0.97	0.96	0.96	0.96
		2	0.98	0.98	0.98	0.96	0.96	0.96
		3	0.99	0.99	0.99	0.97	0.97	0.97
		4	1	1	1	0.98	0.98	0.98
		5	1	1	1	0.98	0.98	0.98
	dataset4	1	0.87	0.83	0.86	0.84	0.79	0.83
		2	0.91	0.89	0.91	0.86	0.83	0.86
		3	0.95	0.94	0.95	0.88	0.85	0.88
		4	0.97	0.97	0.97	0.88	0.86	0.89
		5	0.98	0.98	0.98	0.89	0.87	0.89
English	dataset1	1	0.87	0.85	0.86	0.84	0.83	0.84
		2	0.96	0.96	0.96	0.92	0.92	0.92
		3	0.97	0.97	0.97	0.92	0.92	0.92
		4	0.99	0.99	0.99	0.94	0.94	0.94
		5	1	1	1	0.94	0.94	0.94
	dataset2	1	0.86	0.85	0.85	0.84	0.83	0.83
		2	0.95	0.94	0.95	0.9	0.9	0.9
		3	0.98	0.98	0.98	0.92	0.92	0.92
		4	0.99	0.99	0.99	0.93	0.93	0.93
		5	0.99	0.99	0.99	0.93	0.93	0.93
	dataset3	1	0.99	0.99	0.99	0.98	0.98	0.98
		2	0.99	0.99	0.99	0.98	0.98	0.98
		3	0.99	0.99	0.99	0.98	0.98	0.98
		4	1	1	1	0.98	0.98	0.98
		5	1	1	1	0.98	0.98	0.98
	dataset4	1	0.87	0.85	0.86	0.9	0.88	0.9
		2	0.96	0.96	0.96	0.93	0.92	0.93
		3	0.97	0.97	0.97	0.95	0.94	0.95
		4	0.99	0.99	0.99	0.96	0.96	0.96
		5	1	1	1	0.97	0.96	0.97
	dataset5	1	0.93	0.9	0.93	0.9	0.87	0.9
		2	0.94	0.91	0.94	0.92	0.88	0.91
		3	0.95	0.92	0.95	0.92	0.89	0.92
		4	0.96	0.95	0.96	0.92	0.88	0.92
		5	0.98	0.96	0.98	0.93	0.9	0.93
Turkish	dataset1	1	0.79	0.77	0.78	0.75	0.74	0.74
		2	0.9	0.89	0.9	0.85	0.84	0.84
		3	0.97	0.97	0.97	0.89	0.89	0.89
		4	0.99	0.99	0.99	0.91	0.91	0.91
		5	0.99	0.99	0.99	0.92	0.92	0.92
	dataset2	1	0.71	0.66	0.68	0.68	0.63	0.65
		2	0.9	0.89	0.9	0.85	0.84	0.85
		3	0.96	0.95	0.96	0.87	0.87	0.87
		4	0.98	0.98	0.98	0.88	0.89	0.88
		5	0.99	0.99	0.99	0.9	0.9	0.9
	dataset3	1	0.83	0.83	0.83	0.81	0.81	0.81
		2	0.85	0.85	0.85	0.82	0.82	0.82
		3	0.86	0.86	0.86	0.83	0.83	0.83
		4	0.87	0.87	0.87	0.84	0.84	0.84
		5	0.87	0.87	0.87	0.84	0.84	0.84
	dataset4	1	0.87	0.85	0.87	0.84	0.82	0.84
		2	0.91	0.89	0.91	0.87	0.85	0.87
		3	0.94	0.93	0.94	0.88	0.86	0.88
		4	0.97	0.96	0.97	0.89	0.87	0.89
		5	0.98	0.98	0.98	0.9	0.88	0.9
Whole Dataset (with the masking method)		1	0.82	0.47	0.81	0.8	0.46	0.79
		2	0.88	0.7	0.87	0.85	0.67	0.85
		3	0.92	0.82	0.91	0.88	0.78	0.88
		4	0.93	0.87	0.93	0.89	0.83	0.89
		5	0.94	0.90	0.94	0.89	0.85	0.89

4. Conclusion

The study aims to develop a solution to reduce the memory space used in a chatbot architecture that can serve more than one company and has more than one language. In this direction, a masking method has been developed with prior knowledge of the class ranges specific to the problem (firm) to be estimated. By including the masking method in the estimation phase of the trained model, a single model to be used in the chatbot architecture with the technique mentioned above consumes less memory space than more than one model, providing similar performance. For this purpose, a single BERT model containing different classification problem data was compared with other BERT models that were fine-tuned for the problem. The selection of classification problems to be used in the study aimed to emphasize the usability of the developed structure in the outside world by selecting problems in similar areas.

We achieved an average accuracy of 92%. The model, which included all data sets and did not use prior knowledge of the problem, performed 80% estimation. However, with the solution found, the model trained with the data sets of all problems was asked to make predictions with the help of prior knowledge instead of making predictions directly.

While five models trained based on the problem take up 10 GB, a single model trained as a result of combining the datasets takes up only 2 GB. While switching to the single model, we compromised only 1.2% of the accuracy value. In comparison, 80% saving in memory space is achieved. Thanks to the structure created, the number of problems handled and the memory gain change can be seen in Equation 1.

To enrich the work and to enlarge its scope, it can be tried to increase the problem's difficulty level by adding different languages and different data sets. In addition, the results obtained with a different architecture to be established can be evaluated by adding a structure consisting of attentions based on words or sentences after the BERT model by going over the 1.2% loss in the accuracy value. By doing translations into different languages, data sets can be added to other languages , and the language diversity in the study can be increased. Instead of the BERT model used in the study, a narrower analysis can be performed with an LSTM structure with smaller model size.

Declaration of Interest

The authors declare that there is no conflict of interest.

Acknowledgements

An earlier version of this paper was presented at the ICADA 2021 Conference and was published in its Proceedings (Title of the conference paper: “Çok Alanlı Chatbot Mimarilerinde Avantajlı Performans ve Bellek Takası”).

References

[1] P. Muangkammuen, N. Intiruk, and K. R. Saikaew, “Automated Thai-FAQ Chatbot using RNN-LSTM,” 2018 22nd International Computer Science and Engineering Conference (ICSEC), 2018.

[2] S. Ozan and D. E. Tasar, “Auto-tagging of short conversational sentences using natural language processing methods,” 2021 29th Signal Processing and Communications Applications Conference (SIU), 2021.

[3] D.E. Taşar, Ş. Ozan., U. Özdil, M.F.Akca, O. Ölmez, S. Gülüm, S. Kutal, and C. Belhan, “Auto-tagging of Short Conversational Sentences using Transformer Methods”. arXiv preprint arXiv:2106.01735, 2021.

[4] D.E. Taşar, Ş. Ozan., M.F.Akca, O. Ölmez, S. Gülüm, S. Kutal, and C. Belhan,“Çok Alanlı Chatbot Mimarilerinde Avantajlı Performans ve Bellek Takası”, presented at ICADA, Online, 26-28 Nov. 2021.

[5] E. S. Tellez, S. Miranda-Jiménez, M. Graff, D. Moctezuma, R. R. Suárez, and O. S. Siordia, “A simple approach to multilingual polarity classification in Twitter,” Pattern Recognition Letters, vol. 94, pp. 68–74, 2017.

[6] X. Ni, J.-T. Sun, J. Hu, and Z. Chen, “Cross lingual text classification by mining multilingual topics from Wikipedia,” Proceedings of the fourth ACM international conference on Web search and data mining - WSDM '11, 2011.

[7] H. Schwenk and X. Li, “A corpus for multilingual document classification in eight languages”, arXiv preprint arXiv:1805.09821, 2018.

[8] X. Zhou, X. Wan, and J. Xiao, “Attention-based LSTM network for cross-lingual sentiment classification,” Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016.

[9] I. Casanueva, T. Temcinas, D. Gerz, M. Henderson ve I. Vulic, "Efficient Intent Detection with Dual Sentence Encoders," 2020.

[10] Lehmann, Jens, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann et al. "DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia." Semantic web 6, no. 2 (2015): 167-195.

[11] Xingkun Liu, Pawel Swietojanski, and Verena Rieser. "Benchmarking Natural Language Understanding Services for building Conversational Agents." . In Proceedings of the Tenth International Workshop on Spoken Dialogue Systems Technology (IWSDS) (pp. xxx–xxx). Springer, 2019.

[12] Ishant, “Emotions in text,” Kaggle, 18-Nov-2020. [Online]. Available: https://www.kaggle.com/ishantjuyal/emotions-in-text. [Accessed: 12-July-2021].

[13] J. Devlin, M.-W. Chang, K. Lee ve K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.