Deep Neural Language-agnostic Multi-task Text Classifier

Karol Gawron, Michał Pogoda, Norbert Ropiak, Michał Swędrowski, Jan Kocoń
Department of Artificial Intelligence, Wrocław University of Science and Technology, Poland
{karol.gawron,michal.pogoda,norbert.ropiak,michal.swedrowski,jan.kocon}@pwr.edu.pl

***Abstract*—Many publications prove that the creation of a multiobjective machine learning model is possible and reasonable. Moreover, we can see significant gains in expanding the knowledge domain, increasing prediction quality, and reducing the inference time. New developments in cross-lingual knowledge transfer open up a range of possibilities, particularly in working with low-resource languages. With a motivation to explore the latest subfields of natural language processing and their interactions, we decided to create a multi-task multilingual model for the following text classification tasks: functional style, domain, readability, and sentiment. The paper discusses the effectiveness of particular language-agnostic approaches to Polish and English and the effectiveness and validity of the multi-task model.**

***Index Terms*—deep learning, language-agnostic, multi-task text classification**

I. Introduction

The process of machine learning originated as an imitation of the actual human learning process. If we delve into this phenomenon's nature, we can notice that learning more complex operations is preceded by learning minor tasks. For example, to calculate a mathematical equation, a student must first learn addition, subtraction, and multiplication. Intuitively, we can conclude that knowing the operations of addition will make it faster to learn multiplication. A closer look reveals that although the tasks listed above may be of different difficulty levels and may deal with seemingly other problems, some knowledge is common to both tasks. In the case of the mathematical operations mentioned previously, this will be the basic knowledge of the numbers' values.

Another great example is the simultaneous learning of multiple languages from the same language group. Learning Italian is much easier if a student knows French or Spanish, as these languages have similar grammar rules or words. This observation leads us to believe that a similar analogy regarding domain knowledge can be made for machine learning.

Multi-task learning (MTL) is creating a single model dedicated to performing more than one task. In this approach, one model for a given input during a single iteration can assign multiple labels for each basic task.

This work was financed by (1) the National Science Centre, Poland, project no. 2019/33/B/HS2/02814; (2) the Polish Ministry of Education and Science, CLARIN-PL; (3) the European Regional Development Fund as a part of the 2014-2020 Smart Growth Operational Programme, CLARIN – Common Language Resources and Technology Infrastructure, project no. POIR.04.02.00-00C002/19; (4) the statutory funds of the Department of Artificial Intelligence, Wrocław University of Science and Technology.

One of the potential benefits is that the context for each task is generalized and thus extended in comparison to single-task models. The comprehensive information about the subject may potentially result in increasing the prediction quality of the model. Another important aspect is the reduction of inference time and resource consumption. Solving tasks one by one with dedicated models can be more time-consuming (or require more resources to parallelize the computation) than a single iteration of a multi-task model.

Multi-task approaches are becoming increasingly common in natural language processing (NLP). To process a text, we need to obtain an appropriate representation for it. For several years, the standard proposed by Mikolov [1] has been the creation of vector embeddings. Nowadays, state-of-the-art approaches often rely on large pre-trained models, such as bidirectional encoder representations from transformers – BERT [2] or language-agnostic sentence representations – LASER [3]. With hundreds of millions of parameters trained on large datasets, such models achieve the best results for many NLP tasks [4].

Researchers highlight the remarkable value of low-resource languages [5]. Thanks to language-agnostic models, solutions and datasets developed for almost any language can be applied cross-linguistically, significantly reducing the potential cost of creating dedicated datasets or language-specific tools. This could be an essential factor for business and science, allowing many areas of NLP in low-resource languages to be explored at a significantly better level. Not surprisingly, this topic has been overgrown in recent years, driven mainly by IT industry leaders such as Google [6] and Facebook [3].

In our solution, we decided to compare the current methods using two languages – resource-rich English and low-resource Polish. It allows us to determine changes in prediction quality depending on the language's resource richness for a particular task. When defining the experiment's scope, we chose four text classification tasks: sentiment, domain, functional style, and readability. The difficulty of each task is differentiated and defined by datasets.

Given the ongoing development of the concerned areas, we have made the following contributions:

- Determine the performance of each language-agnostic approach on each of the selected datasets for resource-rich and low-resource languages.
- Examine the effects of applying the multi-task model and compare it to task-specific models regarding prediction

2375-9259/21/$31.00 ©2021 IEEE
DOI 10.1109/ICDMW53433.2021.00023

quality and time consumed per task.
- Demonstrate that the use of multi-task learning can improve performance for an unbalanced sentiment dataset (a subset of AmazonReviews in our case).

II. Related Work

Multi-task learning is becoming increasingly popular in various artificial intelligence fields, particularly NLP or computer vision. The paper [7] presents the main ideas, i.e., what multi-task learning is, why it works, and the critical problem of determining related tasks. It also shows the performance of multi-task learning on shallow neural networks and compares them with single-task networks.

Architectures and solutions for real-world applications follow this. An example is a paper on speech synthesis [8], showing the small gain of using a common data representation for different tasks and using stacked bottleneck features.

Another exciting solution was using multi-task learning to improve the generalization of models for sentiment classification for datasets with various domains [9]. For this problem, the sentiment classifier was split into two components: a main independent one, which is used to gain general sentiment knowledge, and a side domain-dependent one. Besides, a domain similarity graph is proposed to capture the similarities between different domains and regularize the domain-specific sentiment classifier parameters.

A further idea [10] of applying multi-task to NLP is to predict jointly such classes as part-of-speech tags, chunks, named entity tags, semantic roles, semantically similar words, and the likelihood that the sentence makes sense (language modeling). They used trainable embedding vectors, which are then passed through a TDNN (Time-Delay Neural Network) using convolutional layers. An exciting addition was semi-supervised learning to extend the dataset using publicly available data from the Internet.

Research papers [11], [12] provide an overview of what has been done in multi-task learning and classify the methods into different categories such as hard/soft parameter sharing, feature learning, task clustering, and task relation approaches.

When looking for multi-task text classification solutions, there are many research papers in recent years. One of them [13] introduces a new Gate Multi-Task Convolutional Neural Network (CNN) architecture with separate subnets for each task to utilize all datasets. It uses a gate mechanism to select and share these features between subnetworks.

Another idea is to combine representations based on single tasks. They used word embeddings for binary classification of sentiment and sarcasm [14], followed by sentence embeddings using GRU gates. Finally, the sentence embedding vectors are combined using a neural tensor network (NTN). The results show that the combined training of related tasks such as sarcasm and sentiment prediction results better than single-task models.

In one of the papers on stacked ensemble models [15], the presented approach assumes a combination of Long Short-Term Memory (LSTM), CNN, Gated Recurrent Unit (GRU), and feature-based model based on Support Vector Regression (SVR). All components are fused using a three-layer multilayer perceptron (MLP) network. The author in his work undertakes the tasks of prediction financial sentiment for short texts originated from news sites and microblogs as well as prediction of emotion intensity on tweets. The study has shown the superiority of the ensembled approach over using single models.

A study on depression level estimation [16] looked at a multiparameter architecture, evaluating the model with fully shared parameters against a model which shares only some of the parameters. The tests showed that sharing all parameters between tasks produces better results for most cases.

Some solutions improve the primary task by creating side tasks. A support task such as negative supervision [17] can generate different representations for each class to enhance the text classification results.

However, multi-task learning is not only limited to NLP. There are also studies in computer vision [18], i.e., classifying the artist, the type/style, the object's material, or predicting the year the object was made. An intriguing part of this research work was the integration of weights for each loss function. The loss function for the regression had about ten times larger values than the loss function for multi-label classification.

In summary, many research papers on multi-task learning have proposed new architectures that have matched or beaten previous state-of-the-art or single-task models. Many used the weighted average of single loss functions for each task, manually choosing weights or setting equal values as $1/n$ (n - number of tasks). There is more about how we determined the weights in Section V-A.

III. Datasets

Our research started from the Polish language datasets, where we tried to select different text classification tasks, finally deciding on three: a functional style, domain, and sentiment. The selection of corpora from the resource-rich English language was carried out to match them with the already selected Polish language datasets. The tasks matched each other as closely as possible. We matched analogous tasks for the Polish and English corpora (domain and sentiment classification). In the case of functional styles, we used readability level classification as a supplementary task.

A. Polish Corpus of Wrocław University of Technology

Polish Corpus of Wrocław University of Technology (KPWr) [19]–[21] is a corpus of both written and spoken texts that have been semi-automatically and manually annotated at multiple semantic and grammatical levels. Additionally, metadata such as text domain, keywords, text type, and functional style have been manually assigned to the documents. In this work, we use the domain and functional style labels.

The functional style is a way of expression that depends on the utterance's content, purpose, and function. KPWr documents are annotated with functional style classes: *scientific, press, literary, rhetorical, legal, colloquial*.

TABLE I
CHARACTERISTICS OF DATASETS. *KPWR DOMAIN CLASSES: SOURCES, BIBLIOGRAPHIES, META-DESCRIPTION; CULTURE AND ARTS; GEOGRAPHY AND PLACES; HEALTH AND FITNESS; HISTORY AND EVENTS; MATHEMATICS AND LOGIC (FORMAL SCIENCES); LIFE SCIENCES; PEOPLE AND "THE SELF"; PHILOSOPHY AND THINKING; RELIGION AND BELIEF SYSTEMS; SOCIETY, SOCIAL, AND HUMAN SCIENCES; TECHNOLOGY AND APPLIED SCIENCES. **THE TASK IS MULTI-LABEL, SO THE VALUES DO NOT ADD UP TO 100% BUT REPRESENT THE PERCENTAGE OF DOCUMENTS ANNOTATED WITH A GIVEN CLASS.

Lang.	Dataset	Task	Size [docs]	Classes	Classes in examples [%]
PL	KPWr	functional style	1628	scientific, press, literary, rhetorical, legal, colloquial	34.1/35.1/9.3/6.3/8.0/7.2
PL	KPWr	domain	1628	*12 domain labels (listed in caption)	**5.0/30.0/6.4/2.6/35.9/0.9 /5.3/9.3/0.9/4.5/60.0/22.5
ENG	OneStopEnglish	readability	567	elementary, intermediate, advanced	33.3/33.3/33.3
PL	PolEmo 2.0	sentiment	8216	positive, negative, ambivalent, neutral	27.8/37.9/19.5/14.8
ENG	subset of AmazonReviews	sentiment	2001	rating: 1, 2, 3, 4, 5	2.1/2.4/7.5/20.3/67.6
ENG	subset of AG's News	domain	2000	World, Sports, Business, Sci/Tech	25.0/25.0/25.0/25.0

Unlike the other single tasks in this paper, the domain classification in KPWr is multi-label, i.e., each document has assigned at least one (often several) of the 12 subject labels: *Sources, bibliography, meta-description; Culture and arts; Geography and places; Health and fitness; History and events; Mathematics and logic (formal sciences); Life sciences; People and "the self"; Philosophy and thinking; Religion and belief systems; Society and the social and human sciences; Technology and applied sciences*

B. OneStopEnglish corpus

OneStopEnglish [22] is a corpus containing texts from the website onestopenglish.com, more precisely newspaper articles transcribed by teachers to correspond to 3 different levels of text complexity. This means that each document is available in 3 versions: elementary (the easiest to read); intermediate (containing more complex words); and advanced (closest to the original, the most complex).

We use this corpus as a text readability classification task, using documents as examples of *elementary, intermediate, and advanced* classes.

C. PolEmo 2.0

PolEmo 2.0 [23], [24] is a Polish corpus containing 8,216 reviews consisting of 57,644 sentences. Annotations of the text are at both the sentence level and the document level. Texts originate from four different sources representing four different domains: hotels, products, school, and medicine. PolEmo 2.0 is an integral part of the most popular Polish language benchmark KLEJ [25], which emphasizes its importance.

We can distinguish four labels representing the different sentiment values of a text. Classes and their distributions are described in Table I. Although the dataset consists of reviews, labels were assigned by independent annotators (text recipients) without taking into account the author's opinion (rating) during the final evaluation. A psychologist and a linguist annotated reviews. Another independent annotator solved disagreements [23]. The released collection has already been split into test (820 samples), validation (823 samples), and training (6573 samples) subsets.

D. AmazonReviews

The original Amazon product dataset contains 142.8 million product reviews given by users of the online shopping platform Amazon between 1994 and 2014. It is a well-known source of data, especially in recommendation systems such as the works of Ruining He [26] and Julian McAuley [27].

To bring the collection size in line with the other datasets, 2001 reviews from the *Musical Instruments* subset were selected. The samples were established in the process of stratified random sampling and divided into train/test/valid sets of 80%/10%/10%, respectively.

For this collection, the text's sentiment corresponds to the rating given by the user on Amazon. The ratings range from 1 to 5, where 1 is the most negative sentiment and 5 the most positive one. As we can see in Table I, the dataset is highly unbalanced – there is a significantly higher number of positive reviews (ratings 4 and 5). In other works such as [28] the authors often decide to simplify the task by aggregating classes into smaller clusters. We decided to leave the original annotations to make this task relatively complex.

E. AG's News Topic Classification Dataset

AG's News Topic Classification Dataset is a collection constructed by Xiang Zhang from AG News [1]. Initially, AG News was a collection of over 1 million articles from a variety of news sources. The texts were collected using an academic news search engine called ComeToMyHead. The dataset contains metadata attributes such as: source, URL, title, image, category, description, rank, publication date, and video.

AG's News Topic Classification Dataset was created and used as a text classification benchmark in the paper [29]. The authors extracted about 32,000 articles that were tagged with one of the four most popular topics: *World (0), Sports (1),*

[1] groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

Business (2), Sci/Tech (3). They also extracted the most critical columns, i.e., the article's title, content, and topic label.

IV. Models

To obtain a single multi-task model for both Polish and English texts, we focused on trained language-agnostic models that would allow us to obtain embeddings for each document. To obtain language-agnostic representations, we used the selected state-of-the-art models in their original form (with no fine-tuning). We used these embeddings to train multiple variants of MLP models, as described in Section V.

A. LASER

LASER [3] stands for Language-Agnostic Sentence Representation, and it is a framework introduced by Intelligence Research Lab in 2019. Artetxe and Schwenk highlighted that this tool was created to obtain universal language-agnostic embeddings, which should be especially significant in terms of development for low-resource languages. This model is based on sequence-to-sequence autoencoder architecture, where the encoder is BiLSTM neural network, and the decoder is LSTM neural network. It is worth noticing that only a single encoder and decoder are used, even though in the process of training, 93 different languages are involved. At the end of the learning process, the decoder is discarded to obtain language-agnostic sentence representation in the form of a 1024 dimensional vector from an encoder's output. Assuming that two sentences differ in language but have the same meaning, they will be interpreted as two adjacent points of a common cross-language multidimensional space when processed by LASER. Using this property, we obtain language-independent representations of the text.

B. LaBSE

In 2020, Google AI presented LaBSE [6] (Language agnostic BERT Sentence Encoder) - effective cross-lingual sentence embeddings. LaBSE arose as a result of training a bidirectional dual encoder with additive margin softmax over 6B translation pairs in over 112 languages. The authors emphasize much better performance on less popular languages compared to previous language-agnostic solutions. Both texts are given as input to the dual-encoder model. The training process is performed to make the outputs as similar as possible. The cosine function determines the vector space similarity. The loss function is a sum of the source-to-target and target-to-source loss functions. Feng in his research associates a pre-trained encoder to Masked Language and Translation Language Models on both monolingual data and bilingual translation pairs. Source and target sentences were encoded with a BERT base encoder imposing the maximal sentence length to 512 tokens (BERT architecture constraints).

Similar to the LASER, the output of a model is a language-agnostic representation of a text. A 768-dimensional vector represents each sentence. The evaluation presented in the paper shows a clear advantage over other language-agnostic approaches (LASER and mUSE) and multilingual BERT (however, there is a big difference in the size of vocabulary in favor of LaBSE) in tasks such as finding a translation among a set of sentences from a different language. We used an implementation provided by Sentence Transformers framework[2].

C. distilUSE

The last considered architecture is distilUSE, a model obtained by multilingual knowledge distillation of mUSE [30], a method presented in [31]. This model requires a corpus of sentence pairs with identical meanings but in different languages. One of the languages must be supported by a so-called *teacher* model. It is based on training the *student* model so that pairs of corresponding sentences from different languages have output vectors encoded by the student as close as possible to the output vector of the sentence in English from the teacher model, i.e., ideally, if the output vectors of sentences with the same meaning from different languages are identical. This results in aligned vectors in a shared space across languages.

Specifically, we used the *distiluse-base-multilingual-cased-v2* model available within Sentence Transformers[2]. The mUSE model was used as the teacher model, while the DistilBERT multilingual model acted as the student. Which according to the notation in [31] represents *(DistilBERT <- mUSE)*. Therefore, the output vectors of the disitilUSE model are very close to those of the original mUSE model.

V. Evaluation

We conducted a series of experiments to answer the following questions:

- Which language-agnostic text representation works best for each of the given problems?
- How does combining multiple tasks into one affect model performance and runtime?
- Whether the use of a multi-task model can significantly reduce the total inference time while maintaining prediction quality?

A. Setup

The work associated with our experiment can be broken down into two stages. The first phase involved working with task-specific models. For each dataset, we obtained all three types of agnostic embedding vectors. Subsequently, for each configuration thus obtained, we tested the prediction quality of the classifiers in the form of different MLPs depicted in Table II. At the end of phase one, we collated the results to determine the best task-specific models for each given task.

In the second phase, we combined the datasets of all training and validation data into one. Missing labels for each task were assigned to the best task-specific model of that particular task. This ensured that each text unit from all examined datasets had a label for each task. The combined dataset was used to train a multi-task model. An important factor in the training process

[2]www.sbert.net/

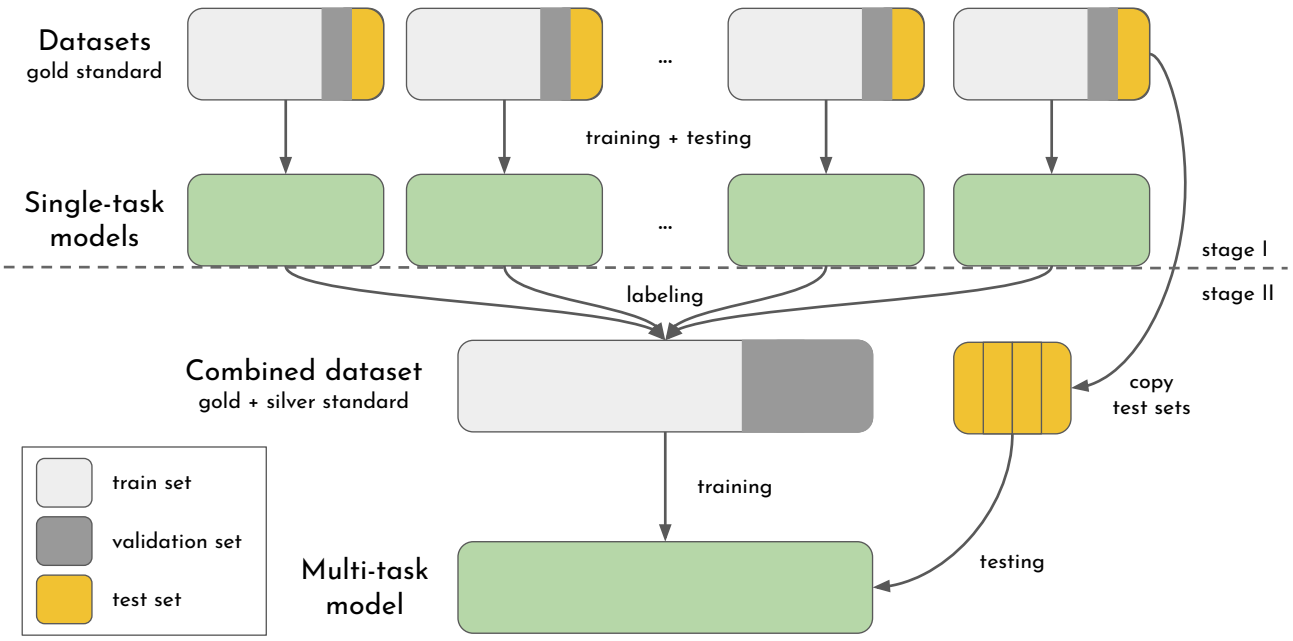

Fig. 1. The pipeline of experiments has two stages: in the first stage, single-task models are trained and tested. A combined training and validation set is created in the second stage, and a multi-task model is trained and tested.

$$\text{Loss} = \sum_{t=1}^{n} w_t L_t \tag{1}$$

Equation 1. L is loss function, and t is a single task, n is a number of tasks, and w stands for single-task loss weight.

was the selection of the loss function. In line with the paper [32], we used the weighted combined loss described in (1). An essential factor affecting the quality of learning is the value of the loss function for each task. According to Strezoski [18], it is essential to keep values in the same order of magnitude. Therefore, we assigned the weight of each single-task loss as a value inversely proportional to the loss function's final value on the validation dataset for the best task-specific model.

Finally, we compared the multi-task model's performance versus the single-task model on each test set corresponding to a specific task. The whole course of action is visualized in Figure 1.

B. Evaluating single-task models

For each task, we ran each MLP configuration (Table II) ten times (resulting in 900 runs), the results shown in all tables are the average of these runs, and metrics were calculated on the test sets. The exact configurations are shown in Table III, where they all use the Adam optimizer with a learning rate set to 10^{-3}. The input and output layers were adjusted according to the embeddings' length and the number of classes, respectively. As a loss, for single-label tasks, we used cross-entropy, while for the multi-label task ([KPWr | domain] only), we used binary cross-entropy.

TABLE II
MLP CONFIGURATIONS.

no.	Hidden units	Batch size	Dropout
#1	128	200	0.3
#2	128 | 64	200	0.3
#3	768 | 64	256	0.8
#4	768 | 64	256	0.4
#5	512 | 256 | 128	256	0.5

We used the F1-macro metric from the validation set to select the best single-task models to prevent *information leakage*. We checked the best MLP model for each text representation obtained with the language-agnostic models. The single-task models' results can be found in Table III (left side of each column). In each task, MLPs using LaBSE embeddings performed best. For the tasks [KPWr | domain] and [OneStop. | read.], respectively, configurations #5 and #2 performed best. In all other tasks, configuration #1 scored best.

C. Evaluating multi-task model

As with the single-task model, for the multi-task models, we checked each configuration (Table II) for each text representation model (resulting in 150 runs). The output of the multi-task network consists of concatenated one-hot vectors for single tasks. Loss is calculated as described in (1), and F1

TABLE III
F1-MACRO FOR THE BEST SINGLE-TASK AND MULTI-TASK MLP MODELS ON SELECTED EMBEDDINGS.

		LaBSE		LASER		distilUSE	
Dataset	**Task**	single-task	multi-task	single-task	multi-task	single-task	multi-task
KPWr	fun. style	75.48	72.76	74.39	72.72	69.60	64.63
OneStop.	read.	50.14	51.41	30.41	41.75	35.13	38.45
KPWr	domain	45.76	39.57	38.85	32.53	45.27	36.75
AG's	domain	84.75	83.21	75.61	74.70	84.46	83.14
PolEmo	sentiment	83.57	83.21	80.25	78.59	74.94	73.52
Amazon.	sentiment	23.68	26.19	18.38	22.29	18.80	23.75
	Avg.	**60.56**	**59.39**	52.98	53.76	54.70	53.37

TABLE IV
COMPARISON OF THE BEST SINGLE-TASK MODELS AND BEST MULTI-TASK MODEL.
STATISTICALLY SIGNIFICANT DIFFERENCES (FROM SINGLE-TASK TO MULTI-TASK) ARE MARKED IN BOLD.

		Test accuracy		Test F1-macro	
Dataset	**Task**	single-task	multi-task	single-task	multi-task
KPWr	functional style	79.88	**77.54 (-2.34)**	75.48	**72.76 (-2.72)**
OneStop.	readability	50.53	51.40 (+0.87)	50.14	51.41 (+1.27)
KPWr	domain	91.94	91.48 (-0.46)	45.76	**39.57 (-6.19)**
AG's	domain	84.70	**81.85 (-2.85)**	84.75	**81.99 (-2.76)**
PolEmo	sentiment	83.40	83.17 (-0.23)	83.57	83.21 (-0.36)
Amazon.	sentiment	66.25	65.80 (-0.45)	23.68	**26.19 (+2.51)**
	Avg.	76.12	**75.21 (-0.91)**	60.56	**59.19 (-1.37)**

metrics and accuracy are calculated on the part of the vector that corresponds to a given task.

We used the F1-macro validation sum for all tasks on the validation set from the combined dataset for evaluation. The test sets were the same for single-task models (and unchanged), allowing for direct comparison. The best results were obtained for configuration #1 using LaBSE embeddings, as shown in Table III. This is the most straightforward configuration tested, and despite the complexity of the task, i.e., the multiplicity of classes, it achieved the best results.

We compared the best multi-task model's performance and the best single-task model, selecting them according to the F1-macro results on the validation set. It is worth noting that the multi-task model was learned on a combined set, labeled in part by precisely these single-task networks.

We presented the average scores for each of the top models in Table IV, where we indicated the statistically significant differences[3] in the multi-task model metrics. Looking at the F1 scores, the performance differences are significant in 4 of the 6 tasks, where 3 of the tasks are unfavorable to the multi-task model. The most significant decrease is seen in the [KPWr | domain] task, for which the best single-task network had the most complex architecture #5.

To compare the runtime of a single multi-task model and 6 single-task models for all tasks, we conducted a test where we measured the inference time of a model (or a set of models) on embeddings from the training set, repeated 10 times ($10 \cdot 11522$ examples). Additionally, we repeated each run 10 times. The results, in the form of prediction times, performed on the available hardware[4], are as follows: single-task models: $74.20 \pm 0.82s$, multi-task model: $11.44 \pm 0.04s$;. On a large scale, this represents more than 6-fold time savings (which can make a difference in production environments) with a low cost in the quality of results.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have prepared a pipeline to compare multilingual single-task models with multi-task ones. To do so, we tested multiple language-agnostic models and different sets of hyperparameters on several popular classification tasks: text-domain, functional style, or sentiment. The LaBSE model was unbeatable for every hyperparameters' value and all tasks in the single-task testing process. In the multi-task testing, we found that although the differences were statistically significant[3], they were relatively small. One of the critical gains of using a multi-task model is the low label inference time for all considered tasks compared to the sequential processing of single-task models. In the case we discussed, the six single tasks' output produced 6.5 slower than the multiobjective approach. This shows how much can be gained, especially from a business point of view. We reduce time and the amount of space the model takes up (assuming that, as in our case, the classifier architectures are similar).

In future work, we plan to extend these models with additional modules and check out other architectures like Caps-Nets or see how tasks can be grouped to improve the performance further. We would also like to test the presented architecture on subjective tasks (including sentiment and emotion recognition) in a personalized approach (the model makes decisions in the context of a specific user), by extending

[3] We checked and compared the results of the single-task and multi-task models with a paired t-test, making sure the distributions were normal with the Shapiro-Wilk test.

[4]MB: ASRock X470 Master SLI; CPU: AMD Ryzen 9 3900X 12-Core Processor; RAM: 64 GB (4x16Gb, 2133 MT/s DDR4)

recently proposed approaches [33]–[35]. We noticed the much lower performance of the multi-task model on tasks where the single model classifier's architecture differs significantly from the proposed multi-task classifier. This led us to the next idea of identifying suitable tasks for being combined in a multi-task approach based on similar architectures of classifiers in single-task models. Perhaps it is not necessarily a good idea to solve all six tasks with a single model, and two smaller multi-task models, handling these tasks together would still offer a reasonable time decrease while not losing prediction quality.

REFERENCES

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

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

[3] M. Artetxe and H. Schwenk, "Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 597–610, 2019.

[4] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," *arXiv preprint arXiv:1804.07461*, 2018.

[5] C. Zhu, H. Yu, S. Cheng, and W. Luo, "Language-aware interlingua for multilingual neural machine translation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1650–1655, 2020.

[6] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic bert sentence embedding," *arXiv preprint arXiv:2007.01852*, 2020.

[7] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[8] Z. Wu, C. Valentini-Botinhao, O. Watts, and S. King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4460–4464, IEEE, 2015.

[9] F. Wu and Y. Huang, "Collaborative multi-domain sentiment classification," in *2015 IEEE International Conference on Data Mining*, pp. 459–468, IEEE, 2015.

[10] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, pp. 160–167, 2008.

[11] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[12] Y. Zhang and Q. Yang, "An overview of multi-task learning," *National Science Review*, vol. 5, no. 1, pp. 30–43, 2018.

[13] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," *arXiv preprint arXiv:1605.05101*, 2016.

[14] N. Majumder, S. Poria, H. Peng, N. Chhaya, E. Cambria, and A. Gelbukh, "Sentiment and sarcasm classification with multitask learning," *IEEE Intelligent Systems*, vol. 34, no. 3, pp. 38–43, 2019.

[15] M. S. Akhtar, A. Ekbal, and E. Cambria, "How intense are you? predicting intensities of emotions and sentiments using stacked ensemble [application notes]," *IEEE Computational Intelligence Magazine*, vol. 15, no. 1, pp. 64–75, 2020.

[16] S. A. Qureshi, S. Saha, M. Hasanuzzaman, and G. Dias, "Multitask representation learning for multimodal estimation of depression level," *IEEE Intelligent Systems*, vol. 34, no. 5, pp. 45–52, 2019.

[17] S. Ohashi, J. Takayama, T. Kajiwara, C. Chu, and Y. Arase, "Text classification with negative supervision," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 351–357, 2020.

[18] G. Strezoski and M. Worring, "Omniart: multi-task deep learning for artistic data analysis," *arXiv preprint arXiv:1708.00684*, 2017.

[19] M. Oleksy, M. Marcińczuk, M. Maziarz, T. Bernaś, J. Wieczorek, A. Turek, D. Fikus, M. Wolski, M. Pustowaruk, J. Kocoń, and P. Kędzia, "Polish Corpus of Wrocław University of Technology 1.3," 2019. CLARIN-PL digital repository.

[20] B. Broda, M. Marcińczuk, M. Maziarz, A. Radziszewski, and A. Wardyński, "KPWr: Towards a free corpus of Polish," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, (Istanbul, Turkey), pp. 3218–3222, European Language Resources Association (ELRA), May 2012.

[21] M. Marcińczuk, M. Oleksy, and J. Kocoń, "Inforex — a collaborative system for text corpora annotation and analysis," in *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, (Varna, Bulgaria), pp. 473–482, INCOMA Ltd., Sept. 2017.

[22] S. Vajjala and I. Lučić, "Onestopenglish corpus: A new corpus for automatic readability assessment and text simplification," in *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pp. 297–304, 2018.

[23] J. Kocoń, P. Miłkowski, and M. Zaśko-Zielińska, "Multi-level sentiment analysis of PolEmo 2.0: Extended corpus of multi-domain consumer reviews," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pp. 980–991, 2019.

[24] J. Kocoń, P. Miłkowski, and K. Kanclerz, "Multiemo: Multilingual, multilevel, multidomain sentiment analysis corpus of consumer reviews," in *International Conference on Computational Science*, pp. 297–312, Springer, 2021.

[25] P. Rybak, R. Mroczkowski, J. Tracz, and I. Gawlik, "KLEJ: comprehensive benchmark for Polish language understanding," *arXiv preprint arXiv:2005.00630*, 2020.

[26] R. He and J. McAuley, "Ups and downs," *Proceedings of the 25th International Conference on World Wide Web*, Apr 2016.

[27] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel, "Image-based recommendations on styles and substitutes," 2015.

[28] T. U. Haque, N. N. Saber, and F. M. Shah, "Sentiment analysis on large scale amazon product reviews," in *2018 IEEE international conference on innovative research and development (ICIRD)*, pp. 1–6, IEEE, 2018.

[29] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *arXiv preprint arXiv:1509.01626*, 2015.

[30] M. Chidambaram, Y. Yang, D. Cer, S. Yuan, Y.-H. Sung, B. Strope, and R. Kurzweil, "Learning cross-lingual sentence representations via a multi-task dual-encoder model," *ACL 2019*, p. 250, 2019.

[31] N. Reimers and I. Gurevych, "Making monolingual sentence embeddings multilingual using knowledge distillation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4512–4525, 2020.

[32] L. Xiao, H. Zhang, and W. Chen, "Gated multi-task network for text classification," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 726–731, 2018.

[33] J. Kocoń, A. Figas, M. Gruza, D. Puchalska, T. Kajdanowicz, and P. Kazienko, "Offensive, aggressive, and hate speech analysis: From data-centric to human-centered approach," *Information Processing & Management*, vol. 58, no. 5, p. 102643, 2021.

[34] K. Kanclerz, A. Figas, M. Gruza, T. Kajdanowicz, J. Kocon, D. Puchalska, and P. Kazienko, "Controversy and conformity: from generalized to personalized aggressiveness detection," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, (Online), pp. 5915–5926, Association for Computational Linguistics, Aug. 2021.

[35] P. Milkowski, M. Gruza, K. Kanclerz, P. Kazienko, D. Grimling, and J. Kocon, "Personal bias in prediction of emotions elicited by textual opinions," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, (Online), pp. 248–259, Association for Computational Linguistics, Aug. 2021.