Domain Adaptation in Multilingual and Multi-Domain Monolingual Settings for Complex Word Identification

George-Eduard Zaharia, Răzvan-Alexandru Smădu
Dumitru-Clementin Cercel, Mihai Dascalu
University Politehnica of Bucharest, Faculty of Automatic Control and Computers
{george.zaharia0806, razvan.smadu}@stud.acs.upb.ro
{dumitru.cercel, mihai.dascalu}@upb.ro

Abstract

Complex word identification (CWI) is a cornerstone process towards proper text simplification. CWI is highly dependent on context, whereas its difficulty is augmented by the scarcity of available datasets which vary greatly in terms of domains and languages. As such, it becomes increasingly more difficult to develop a robust model that generalizes across a wide array of input examples. In this paper, we propose a novel training technique for the CWI task based on domain adaptation to improve the target character and context representations. This technique addresses the problem of working with multiple domains, inasmuch as it creates a way of smoothing the differences between the explored datasets. Moreover, we also propose a similar auxiliary task, namely text simplification, that can be used to complement lexical complexity prediction. Our model obtains a boost of up to 2.42% in terms of Pearson Correlation Coefficients in contrast to vanilla training techniques, when considering the CompLex from the Lexical Complexity Prediction 2021 dataset. At the same time, we obtain an increase of 3% in Pearson scores, while considering a cross-lingual setup relying on the Complex Word Identification 2018 dataset. In addition, our model yields state-of-the-art results in terms of Mean Absolute Error.

1 Introduction

The overarching goal of the complex word identification (CWI) task is to find words that can be simplified in a given text (Paetzold and Specia, 2016b). Evaluating word difficulty represents one step towards achieving simplified, which in return facilitates access to knowledge to a wider audience texts (Maddela and Xu, 2018). However, complex word identification is a highly contextualized task, far from being trivial. The datasets are scarce and, most of the time, the input entries are limited or cover different domains/areas of expertise. Therefore, developing a robust and reliable model that can be used to properly evaluate the complexity of tokens is a challenging task. Table 1 showcases examples of complex words annotations from the CompLex LCP (Shardlow et al., 2020, 2021b) and English CWI (Yimam et al., 2018) datasets employed in this work.

	Domain	Text
CompLex LCP Dataset	Bible	But let each man test his own work, and then he will take pride in himself and not in his neighbor.
	Biomedical	A genome database search revealed orthologs of ADAM11, ADAM22 and ADAM23 genes to exist in vertebrates such as mammals, fish, and amphibians, but not in invertebrates.
	Europarl	They also allow for easy compensation for the thousands of accidents involving vehicles from more than one Member State.
English CWI Dataset	Wikipedia	Normally, the land will be passed down to future generations in a way that recognizes the community's traditional connection to that country.
	WikiNews	The JAS 39C Gripen crashed onto a runway at around 9:30 am local time (02:30 UTC) and exploded, closing the airport to commercial flights.
	News	The car has been removed from the scene for forensic technical examination.

Table 1: Examples of complex words annotated for each of the domains from CompLex LCP and CWI datasets. The shades indicate the complexity; the darker the shade, the more complex the sequence of words. Best viewed in color.

Nevertheless, certain training techniques and auxiliary tasks help the model improve its generalization abilities, forcing it to focus only on the most relevant, general features (Schrom et al., 2021). Techniques like domain adaptation (Ganin et al., 2016) can be used for various tasks, with the purpose of selecting relevant features for follow-up processes. At the same time, the cross-domain scenario can be transposed to a cross-lingual setup, where the input entries are part of multiple available languages. Performance can be improved by

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 70 - 80

also employing the power of domain adaptation, where the domain is the language; as such, the task of identifying complex tokens can be approached even for low resource languages.

We propose several solutions to improve the performance of a model for CWI in a cross-domain or a cross-lingual setting, by adding auxiliary components (i.e., Transformer (Vaswani et al., 2017) decoders, Variational Auto Encoders - VAEs (Kingma and Welling, 2014)), as well as a domain adaptation training technique (Farahani et al., 2021). Moreover, we use the domain adaptation intuition and we apply it in a multi-task adversarial training scenario, where the main task is trained alongside an auxiliary one, and a task discriminator has the purpose of generalizing task-specific features.

We summarize our main contributions as follows:

- Applying the concept of domain adaptation in a monolingual, cross-domain scenario for complex word identification;
- Introducing the domain adaptation technique in a cross-lingual setup, where the discriminator has the purpose to support the model extract only the most relevant features across all languages;
- Proposing additional components (i.e., Transformer decoders and Variational Auto Encoders) trained alongside the main CWI task to provide more meaningful representations of the inputs and to ensure robustness, while generating new representations or by tuning the existing ones;
- Experimenting with an additional text simplification task alongside domain/language adaptation, with the purpose of extracting cross-task features and improving performance.

2 Related Work

Domain Adaptation. Several works employed domain adaptation to improve performance. For example, Du et al. (2020) approached the sentiment analysis task by using a BERT-based (Devlin et al., 2019) feature extractor alongside domain adaptation. Furthermore, McHardy et al. (2019) used domain adaptation for satire detection, with the publication source representing the domain. At the same time, Dayanik and Padó (2020) used a technique similar to domain adaptation, this time for political claims detection. The previous approaches consisted of actor masking, as well as adversarial debiasing and sample weighting. Other studies considering domain adaptation included suggestion mining (Klimaszewski and Andruszkiewicz, 2019), mixup synthesis training (Tang et al., 2020), and effective regularization (Vernikos et al., 2020).

Cross-Lingual Domain Adaptation. Chen et al. (2018) proposed ADAN, an architecture based on a feed-forward neural network with three main components, namely: a feature extractor, a sentiment classifier, and a language discriminator. The latter had the purpose of supporting the adversarial training setup, thus covering the scenario where the model was unable to detect whether the input language was from the source dataset or the target one. A similar cross-lingual approach was adopted by Zhang et al. (2020), who developed a system to classify entries from the target language, while only labels from the source language were provided.

Keung et al. (2019) employed the usage of multilingual BERT (Pires et al., 2019) and argued that a language-adversarial task can improve the performance of zero-resource cross-lingual transfers. Moreover, training under an adversarial technique helps the Transformer model align the representations of the English inputs.

Under a Named Entity Recognition training scenario, Kim et al. (2017) used features on two levels (i.e., word and characters), together with Recurrent Neural Networks and a language discriminator used for the domain-adversarial setup. Similarly, Huang et al. (2019) used target language discriminators during the process of training models for low-resource name tagging.

Word Complexity Prediction. Gooding and Kochmar (2019) based their implementation for CWI as a sequence labeling task on Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) networks, inasmuch as the context helps towards proper identification of complex tokens. The authors used 300-dimensional pretrained word embeddings as inputs for the LSTMs. Also adopting a sequence labeling approach, Finnimore et al. (2019) considered handcrafted features, including punctuation or syllables, that can properly identify complex structures.

The same sequence labeling approach can be applied under a plurality voting technique (Polikar, 2006), or even using an Oracle (Kuncheva et al.,

2001). The Oracle functions best when applied to multiple solutions, by jointly using them to obtain a final prediction. At the same time, Zaharia et al. (2020) explored the power of Transformer-based models (Vaswani et al., 2017) in cross-lingual environments by using different training scenarios, depending on the scarcity of the resources: zero-shot, one-shot, as well as few-shot learning. Moreover, CWI can be also approached as a probabilistic task. For example, De Hertog and Tack (2018) introduced a series of architectures that combine deep learning features, as well as handcrafted features to address CWI as a regression problem.

3 Method

3.1 Datasets

We experimented with two datasets, one monolingual - CompLex LCP 2021 (Shardlow et al., 2020, 2021b) - and one cross-lingual - the CWI Shared Dataset (Yimam et al., 2018). The entries of *CompLex* consist of a sentence in English and a target token, alongside the complexity of the token, given its context. The complexities are continuous values between 0 and 1, annotated by various individuals on an initial 5-point Likert scale; the annotations were then normalized.

The *CompLex* dataset contains two types of entries, each with its corresponding subset of entries: a) single, where the target token is represented by a single word, and b) multiple, where the target token is represented by a group of words. While the single-word dataset contains 7,662 training entries, 421 trial entries, and 917 test entries, the multi-word dataset has lower counts, with 1,517 training entries, 99 trial entries, and 184 for testing. At the same time, the entries correspond to three different domains (i.e., biblical, biomedical, and political), therefore displaying different characteristics and challenging the models towards generalization.

The *CWI dataset* was introduced in the CWI Shared Task 2018 (Yimam et al., 2018). It is a multilingual dataset, containing entries in English, German, Spanish, and French. Moreover, the English entries are split into three categories, depending on their proficiency levels: professional (News), non-professional (WikiNews), and Wikipedia articles. Most entries are for the English language (27,299 training and 3,328 validation), while the fewest training entries are for German (6,151 training and 795 validation). The French language does not contain training or validation entries.

3.2 The Domain Adaption Model

The overarching architecture of our method is introduced in Figure 1. All underlying components are presented in detail in the following subsections. Our model combines character-level BiLSTM features (i.e., $\mathcal{F}_t$) with Transformer-based features for the context sentence (i.e., $\mathcal{F}_c$). The concatenated features ($\mathcal{F}_c$+$\mathcal{F}_t$) are then passed through three linear layers, with a dropout separating the first and second. The output is a value representing the complexity of the target word.

Three configurations were experimented. Within **Basic Domain Adaptation**, the previous features are passed through an additional component, the domain discriminator, composed of a linear layer followed by a softmax activation function. A *gradient reversal* layer (Ganin and Lempitsky, 2015) is added between the feature concatenation and the discriminator to reverse the gradients through the backpropagation phase and support extracting general features. The loss function is determined by Equation 1 as:

$$L = L_r - \beta\lambda L_d \tag{1}$$

where L_r is the regression loss, L_d is the general domain loss, β is a hyperparameter used for controlling the importance of L_d, and λ is another hyperparameter that varies as the training process progresses.

The following setups also include the Basic Domain Adaptation training setting.

VAE and Domain Adaptation considers the previous configuration, plus the VAE encoder, that yields the $\mathcal{F}_v$ features, and the VAE decoder, which aims to reconstruct the input. The concatenation layer now contains the BiLSTM and Transformer features, along with the VAE encoder features ($\mathcal{F}_v$), namely $\mathcal{F}_t$+$\mathcal{F}_c$+$\mathcal{F}_v$. The loss function is depicted by Equation 2 as:

$$L = L_r - \beta\lambda L_d + \alpha L_v \tag{2}$$

where, additionally, L_v represents the VAE loss described in Equation 6.

Transformer Decoder and Domain Adaptation adds a Transformer Decoder with the purpose of reconstructing the original input, for a more robust context feature extraction. The loss is denoted by Equation 3 as:

$$L = L_r - \beta\lambda L_d + \alpha L_{dec} \tag{3}$$

where L_{dec} represents the decoder loss described in Equation 9.

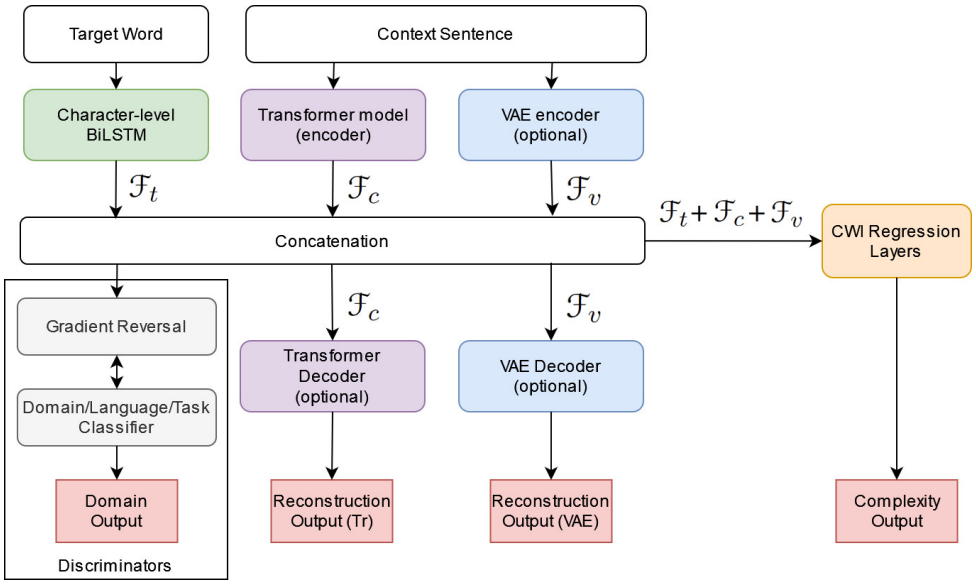

Figure 1: The overarching architecture for the domain adaptation model.

3.2.1 Character-level BiLSTM for Target Word Representation

The purpose of this component is to determine the complexity of the target token, given only its constituent characters. A character-level Bidirectional Long Short-Term Memory (BiLSTM) network receives as input an array of characters corresponding to the target word (or group of words), and yields a representation that is afterwards concatenated to the previously mentioned Transformer-based representations. Each character c is mapped to a certain value obtained from the character vocabulary V, containing all the characters present in the input dataset.

The character sequence is represented as $C^i = [c_1, c_2, \ldots, c_n]$, where n is the maximum length of a target token. C^i is then passed through a character embedding layer, thus yielding the output Emb_{target}. Emb_{target} is then fed to the BiLSTM, followed by a dropout layer, thus obtaining the final target word representation, $\mathcal{F}_t$.

3.2.2 Transformer-based Context Representation

We rely on a Transformer-based model as the main feature extractor for the context of the target word (i.e., the full sentence), considering their superior performance on most natural language processing tasks. The selected model for the first dataset is RoBERTa (Liu et al., 2019), as it yields better results when compared to its counterpart, BERT. RoBERTa is trained with higher learning rates and larger mini-batches, and it modifies the key hyper-parameters of BERT. We employed the usage of XLM-RoBERTa (Conneau et al., 2020), the multilingual counterpart of RoBERTa, now trained on a very large corpus of multilingual texts, for the second cross-lingual task. The features used for our task are represented by the pooled output of the Transformer model. The feature vector $\mathcal{F}_c$ of 768 elements captures information about the context of the target word.

3.2.3 Variational AutoEncoders

We aim to further improve performance by adding extra features via Variational AutoEncoders (VAEs) (Kingma and Welling, 2014) to the context representation for a target word. More specifically for the CWI task, we use the latent vector z, alongside the Transformer and the Char BiLSTM features. Moreover, we also need to ensure that the Encoder representation is accurate; therefore, we consider the VAE encoding and decoding as an additional task having the purpose of minimizing the reconstruction loss.

The VAE consists of two parts, namely the encoder and the decoder. The encoder $g(x)$ produces the approximation $q(z|x)$ of the posterior distribution $p(z|x)$, thus mapping the input x to the latent space z. The process is presented in Equation 4. We use as features the representation z, denoted as $\mathcal{F}_v$.

$$p(z|x) \approx q(z|x) = \mathcal{N}(\mu(x), \sigma(x)) \quad (4)$$

The decoder $f(z)$ maps the latent space to the input space (i.e., $p(z)$ to $p(x)$), by using Equation 5.

$$p(x) = \int p(x|z)p(z)dz \\ = \int \mathcal{N}(f(z), I)p(z)dz \tag{5}$$

Equation 6 introduces the loss function, where D_{KL} represents the Kullback Leibler divergence. Furthermore, $\mathbb{E}_q$ represents the expectation with relation to the distribution q.

$$L(f, g) = \sum_i \{-D_{KL}[q(z|x_i)||p(z)] \\ + \mathbb{E}_{q(z|x_i)}[\ln p(x_i|z)]\} \tag{6}$$

3.2.4 Discriminators

The features extracted by our architecture can vary greatly as the input entries can originate from different domains or languages. Consequently, we introduced a generalization technique to extract only cross-domain features that do not present a bias towards a certain domain. We thus employ an adversarial training technique based on domain adaptation, forcing the model to only extract relevant cross-domain features.

A discriminator acts as a classifier, containing three linear layers with corresponding activation functions. The discriminator classifies the input sentence into one of the available domains. Unlike traditional classification approaches, our purpose is not to minimize the loss, but to maximize it. We want our model to become incapable of distinguishing between different categories of input entries, therefore extracting the most relevant, cross-domain features.

Our architecture is encouraged to generalize in terms of extracted features by the *gradient reversal* layer that reverses the gradients during the backpropagation phase; as such, the parameters are updated towards the direction that maximizes the loss instead of minimizing it.

Three scenarios were considered, each one targeting a different approach towards domain adaptation.

Domain Discriminator. The first scenario is applied on the first dataset, CompLex, with entries only in English, but covering multiple domains. The discriminator has the purpose of identifying the domain of the entry, namely biblical, biomedical or political. The intuition is that, by grasping only cross-domain features, the performance of the model increases on all three domains, instead of performing well only on one, while poorer on the others.

Language Discriminator. The intuition is similar to the previous scenario, except that we experimented with the second multilingual dataset. Therefore, our interest was that our model extracts cross-lingual features, such that the performance is equal on all the target languages.

Task Discriminator. In this scenario, we trained a similar, auxiliary task, represented by text simplification. A task discriminator is implemented to detect the origin of the input entry: either the main task or the auxiliary task (i.e., simplified version). The dataset used for text simplification is represented by BenchLS (Paetzold and Specia, 2016a)[1]. The employed simplification process consists of masking the word considered to be complex and then using a Transformer for Masked Language Modeling to predict the best candidate. The corresponding flow is described in Algorithm 1, while the loss function is presented in Equation 7:

$$L = L_r - \beta\lambda L_{task\_id} + L_{ML} \tag{7}$$

where L_{ML} is the Sparse Categorical Cross Entropy loss.

All previous discriminators use the same loss, namely Categorical Cross Entropy (Zhang and Sabuncu, 2018).

The overall loss consists of the difference between the task loss and the domain/language loss. Moreover, the importance of the latter can be controlled by multiplication with a λ hyperparameter, that changes over time, and a fixed β hyperparameter. The network parameters, θ_p are updated according to Equation 8, where η is the learning rate, L_d is the domain loss, L_r is the task loss and β is the weight for the domain loss. A similar equation for language loss (L_l) is in place for the second dataset, where instead of the domain loss L_d we used the language identification loss L_l, having the same formula.

$$\theta_p = \theta_p - \eta(\frac{\partial L_r}{\partial \theta_p} - \beta\lambda\frac{\partial L_d}{\partial \theta_p}) \tag{8}$$

3.2.5 Transformer Decoder

Our model also considers a decoder to reconstruct the original input, starting from the Transformer representation. The intuition behind introducing

[1] http://ghpaetzold.github.io/data/BenchLS.zip

Algorithm 1: The Multi-Task Adversarial algorithm (Task 1 - lexical complexity prediction; Task 2 - text simplification).

Inputs: Preprocessed dataset, split into batches (x_i, y_i), *i=1,n* (where *n* is the number of batches, x_i are the input features for the target word and the context, and y_i is the complexity);
Outputs: Updated parameters θ_p;
Initialization: Initialize θ_p with random weights;
for *every batch* **do**
    Select entries E1 from Task 1;
    Select entries E2 from Task 2;
    out1 = Apply initial architecture on E1;
    out2 = Apply Masked Language Modeling Transformer on E2;
    F = Combine the features from applying architecture on E1 and E2;
    out_task = Pass F through task discriminator;
    loss1 = L_r(out1, ref1);
    loss2 = L_{ML}(out, ref2);
    task_loss = L_{task_id}(out_task, ref_task);
    loss = loss1+loss2-$\beta\lambda$task_loss;
    Backpropagate loss;
    Update θ_p;
end

this decoder is to increase the robustness of the context feature extraction.

The decoder receives as input the outputs of the hidden Transformer layer alongside an embedding of the original input, which are passed through a Gated Recurrent Unit (GRU) (Chung et al., 2014) layer for obtaining the final representation of the initial input. Additionally, two linear layers separated by a dropout are introduced before obtaining the final representation, $y = \mathcal{F}_d$. The loss is computed by using the Negative Log Likelihood loss between the outputs of the decoder and the original Transformer input id representation of the entries (see Equations 9 and 10).

$$L(x, y) = \sum_{n=1}^{N} l_n \tag{9}$$

$$l_n = -w_{y_n} x_{n,y_n}, \quad w_c = weight[c] \cdot \mathbb{1}\{c \neq ignore\_index\} \tag{10}$$

3.3 Experimental Setup

The optimizer used for our models is represented by AdamW (Kingma and Ba, 2014). The learning rate is set to *2e-5*, while the loss functions used for the complexity task are the L1 loss (Janocha and Czarnecki, 2016) for the CompLex LCP dataset and the Mean Squared Error (MSE) loss (Kline and Berardi, 2005) for the CWI dataset. The auxiliary losses are summed to the main loss (i.e., complexity prediction) and are scaled according to their priority, with a factor of α, where α is set to 0.1 for the VAE loss, and 0.01 for the Transformer decoder and task discriminator losses. The λ parameter used for domain adaptation was updated according to Equation 11:

$$\lambda = \frac{2}{1 + e^{-\gamma\epsilon}} - 1 \tag{11}$$

where ϵ is the number of epochs the model was trained; γ was set to 0.1, while β was set to 0.2. Moreover, each model was trained for 8 epochs, except for the one including the VAE features, which was trained for 12 epochs.

4 Results

4.1 LCP 2021 CompLex Dataset

We consider as baselines two models used for the LCP 2021 competition (Shardlow et al., 2021a), as well as the best-registered score. Almeida et al. (2021) employed the usage of neural network solutions; more specifically, they used chunks of the sentences obtained with Sent2Vec as input features. Zaharia et al. (2021) created models that are based on target and context feature extractors, alongside features resulted from Graph Convolutional Networks, Capsule Networks, and pre-trained word embeddings.

Table 2 depicts the results obtained for the English dataset using domain adaptation and various configurations. "Base" denotes the initial model (RoBERTa + Char BiLSTM) on which we apply domain adaptation, as well as the auxiliary tasks. The domain adaptation technique offers improved performance when applied on top of an architecture, considering that the model learns cross-domain features. The only exception is represented by a slightly lower Pearson score on the model that uses domain adaptation alongside the Transformer decoding auxiliary task (Base + Decoder + DA), with a value of .7969 on the trial dataset, when compared

Table 2: Results on the LCP 2021 English dataset.

Model	Single-Word Target				Multi-Word Target			
	Trial		Test		Trial		Test	
	Pearson	MAE	Pearson	MAE	Pearson	MAE	Pearson	MAE
Almeida et al. (2021)	-	-	.4598	.0866	-	-	.3941	.1145
Zaharia et al. (2021)	.7702	.0671	.7324	.0677	.7227	.0863	.7962	.0754
1st Place, LCP 2021 (Shardlow et al., 2021a)	-	-	**.7886**	**.0609**	-	-	**.8612**	**.0616**
Base (RoBERTa + Char BiLSTM)	.7987	.0654	.7502	.0682	.7565	.0828	.8138	.0739
Base + DA	.8111	.0660	.7569	.0657	**.7900**	**.0724**	.8246	.0699
Base + VAE + DA	.8010	.0658	.7554	.0669	.7919	.0745	.8167	.0761
Base + Decoder + DA	.7969	.0687	.7542	.0704	.7747	.0812	.8252	.0693
Base + Text simplification + DA	**.8170**	**.0648**	.7744	.0652	.7670	.0787	.8285	.0708

* DA = Domain Adaptation; VAE = Variational AutoEncoder; Decoder = Transformer Decoder; Pearson = Pearson Correlation Coefficient; MAE = Mean Absolute Error.

Table 3: Results on the CWI 2018 multilingual validation dataset.

Model	EN-N		EN-WN		EN-W		DE		ES	
	P	MAE	P	MAE	P	MAE	P	MAE	P	MAE
Base (XLM-RoBERTa + Char BiLSTM)	.8517	.0476	.8460	**.0512**	.7640	**.0697**	.7092	.0559	**.6944**	.0635
Base + LA	.8592	.0468	.8431	.0532	.7773	.0702	.6857	**.0551**	.6868	**.0625**
Base + VAE + LA	.8557	**.0463**	.8376	.0527	.7562	.0702	.7026	.0565	.6805	.0628
Base + Decoder + LA	.8492	.0511	.8273	.0569	.7619	.0745	.6823	.0645	.6519	.0725
Base + Text simplification + LA	**.8602**	.0514	**.8555**	.0560	**.7842**	.0716	**.7147**	.0621	.6787	.0688

* LA = Language Adaptation; VAE = Variational AutoEncoder; Decoder = Transformer Decoder; EN-N = English-News; EN-WN = English-WikiNews; EN-W = English-Wikipedia; DE = German; ES = Spanish; P = Pearson Correlation Coefficient; MAE = Mean Absolute Error.

Table 4: Results on the CWI 2018 multilingual test dataset.

Model	EN-N		EN-WN		EN-W		DE		ES		FR	
	P	MAE	P	MAE	P	MAE	P	MAE	P	MAE	P	MAE
Kajiwara and Komachi (2018)	-	.0510	-	.0704	-	.0931	-	.0610	-	.0718	-	.0778
Bingel and Bjerva (2018)	-	-	-	-	-	-	-	.0747	-	.0789	-	**.0660**
Gooding and Kochmar (2018)	-	.0558	-	.0674	-	.0739	-	-	-	-	-	-
Base (XLM-RoBERTa + Char BiLSTM)	.8560	.0461	.8045	.0533	.7205	.0679	**.7405**	**.0540**	.6873	.0619	.5506	.0793
Base + LA	**.8582**	.0466	.8146	**.0513**	.7310	.0700	.6866	.0558	.6809	.0606	.5409	.0842
Base + VAE + LA	.8580	**.0450**	.8060	.0526	.7354	**.0671**	.7131	.0553	**.6912**	**.0595**	**.5559**	.0752
Base + Decoder + LA	.8533	.0509	.7978	.0560	.7124	.0708	.6976	.0653	.6490	.0692	.4663	.0889
Base + Text simplification + LA	.8580	.0502	**.8338**	.0539	**.7420**	.0707	.7230	.0614	.6837	.0671	.5394	.0876

* LA = Language Adaptation; VAE = Variational AutoEncoder; Decoder = Transformer Decoder; EN-N = English-News; EN-WN = English-WikiNews; EN-W = English-Wikipedia; DE = German; ES = Spanish; FR = French; P = Pearson Correlation Coefficient; MAE = Mean Absolute Error.

to the initial .7987 (Base). However, the remaining models improve upon the starting architecture, with the largest improvements being observed for domain adaptation and the text simplification auxiliary task (Base + Text simplification + DA), with a Pearson correlation coefficient on the test dataset of .7744, 2.42% better than the base model. The improved performance can be also seen for the Mean Absolute Error score (MAE = .0652).

While the Transformer decoder auxiliary task does not offer the best performance for the single word dataset, the same architecture offers the second-best performance for the multi-word dataset, with a Pearson score of .8252 compared

to the best one, .8285. The domain adaptation and VAE configuration provide improvements upon the base model (.7554 versus .7502 Pearson), but the VAE does not have an important contribution, considering that the Base + domain adaptation model has a slightly higher Pearson score of .7569.

4.2 CWI 2018 Dataset

We also experimented with a multilingual dataset, where the discriminant is considered to be the language. The baseline consists of three models used from the CWI 2018 competition. The performance is evaluated in terms of MAE; however, we also report the Pearson Correlation Coefficient. First, Kajiwara and Komachi (2018) based their models on regressors, alongside features represented by the number of characters or words and the frequency of the target word in certain corpora. Second, the approach of Bingel and Bjerva (2018) is based on Random Forest Regressors, as well as feed-forward neural networks alongside specific features, such as log-probability, inflectional complexity, or target-sentence similarity; the authors focused on non-English entries. Third, Gooding and Kochmar (2018) approach the English section of the dataset by employing linear regressions. The authors used several types of handcrafted features, including word n-grams, POS tags, dependency parse relations, and psycholinguistic features.

Table 3 presents the results obtained on the multilingual validation dataset and compares the performance of different configurations. The best overall performance in terms of Pearson correlation coefficient is yielded by the Base model (XLM-RoBERTa + Char BiLSTM) alongside the text simplification auxiliary task and the domain adaptation technique (Base + Text simplification + LA), with values of .8602 on English News, .8555 on English WikiNews, as well as .7842 on English Wikipedia and .7147 on German. The best Pearson score for the Spanish language is obtained by the base model, with .6944. The Base + VAE + LA architecture offers improvements over the Base model, but falls behind when compared to the Base + Text simplification + LA model, with Pearson correlation ranging from .8557 on the English News dataset to .6805 on the Spanish dataset.

However, when switching to MAE, the metric used for evaluation in the CWI 2018 competition, the best performance is split between the first three models, namely Base, Base + LA, and Base + VAE + LA. The Base + LA approach yields the best, lowest MAE score on the German and Spanish datasets, while the Base architecture performs the best on English WikiNews and English Wikipedia. The English News achieves the best MAE results from the Base + VAE + LA model.

Nevertheless, the best overall performance is obtained by the Base + VAE + LA model on the test dataset (see Table 4), with dominating Pearson and MAE scores on the Spanish and French languages: 0.6912 Pearson, 0.595 MAE for Spanish, as well as .5559 Pearson, and .0752 MAE for French, respectively. The Base + Text simplification + LA model performs the best in terms of Pearson Correlation Coefficient on the English WikiNews and Wikipedia datasets, with Pearson scores of .8338 and .7420. However, the best MAE scores for the same datasets are generated by the Base + LA model (.0513 English WikiNews) and Base + VAE + LA (.0671 English Wikipedia).

5 Discussions

The domain adaptation technique supports our model to learn general cross-domain or cross-language features, while achieving higher performance. Moreover, jointly training on two different tasks (i.e., lexical complexity prediction and text simplification), coupled with domain adaptation to generalize the features from the two tasks, can lead to improved results.

However, there are entries for which our models were unable to properly predict the complexity score, namely: a) entries with a different level of complexity (i.e. biomedical), and b) entries part of a language that was not present in the training dataset (i.e., French). For the former, scientific terms (e.g., "sitosterolemia"), abbreviations (e.g., "ES"), or complex elements (e.g., "H3-2meK9") impose a series of difficulties for our feature extractors, considering the absence of these tokens from the Transformer vocabulary. The latter category of problematic entries creates new challenges in the sense that it represents a completely new language on which the architecture is tested. However, as seen in the results section, the cross-lingual domain adaptation technique offers good improvements, helping the model achieve better performance on French, even though the initial architecture was not exposed to any French example.

6 Conclusions and Future Work

This work proposes a series of training techniques, including domain adaptation, as well as multi-task adversarial learning, that can be used for improving the overall performance of the models for CWI. Domain adaptation improves results by encouraging the models to extract more general features, that can be further used for the lexical complexity prediction task. Moreover, by jointly training the model on the CWI tasks and an auxiliary similar task (i.e., text simplification), the overall performance is improved. The task discriminator also ensures the extraction of general features, thus making the model more robust on the CWI dataset.

For future work, we intend to experiment with meta-learning (Finn et al., 2017) alongside domain adaptation (Wang et al., 2019), considering the scope of the previously applied training techniques. This would enable us to initialize the model's weights in the best manner, thus ensuring optimal results during the training phase.

Acknowledgments

This research was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS - UEFISCDI, project number TE 70 PN-III-P1-1.1-TE-2019-2209, "ATES - Automated Text Evaluation and Simplification".

References

Raul Almeida, Hegler Tissot, and Marcos Didonet Del Fabro. 2021. C3sl at semeval-2021 task 1: Predicting lexical complexity of words in specific contexts with sentence embeddings. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 683–687.

Joachim Bingel and Johannes Bjerva. 2018. Cross-lingual complex word identification with multitask learning. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pages 166–174.

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2018. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, 6:557–570.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Deep Learning and Representation Learning Workshop*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard 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.

Erenay Dayanik and Sebastian Padó. 2020. Masking actor information leads to fairer political claims detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4385–4391.

Dirk De Hertog and Anaïs Tack. 2018. Deep learning architecture for complex word identification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 328–334.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, and Jianxin Liao. 2020. Adversarial and domain-aware bert for cross-domain sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4019–4028.

Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. 2021. A brief review of domain adaptation. *Advances in Data Science and Information Engineering*, pages 877–894.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135.

Pierre Finnimore, Elisabeth Fritzsch, Daniel King, Alison Sneyd, Aneeq Ur Rehman, Fernando Alva-Manchego, and Andreas Vlachos. 2019. Strong baselines for complex word identification across multiple languages. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 970–977.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky.

2016. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030.

Sian Gooding and Ekaterina Kochmar. 2018. Camb at cwi shared task 2018: Complex word identification with ensemble-based voting. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 184–194.

Sian Gooding and Ekaterina Kochmar. 2019. Complex word identification as a sequence labelling task. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1148–1153.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Lifu Huang, Heng Ji, and Jonathan May. 2019. Cross-lingual multi-level adversarial transfer to enhance low-resource name tagging. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3823–3833.

Katarzyna Janocha and Wojciech Marian Czarnecki. 2016. On loss functions for deep neural networks in classification. *Schedae Informaticae*, 25:49–59.

Tomoyuki Kajiwara and Mamoru Komachi. 2018. Complex word identification based on frequency in a learner corpus. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pages 195–199.

Phillip Keung, Vikas Bhardwaj, et al. 2019. Adversarial learning with contextual embeddings for zero-resource cross-lingual classification and ner. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1355–1360.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2832–2838.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations, ICLR 2015*, San Diego, CA, USA.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations, ICLR 2014*, Banff, AB, Canada.

Mateusz Klimaszewski and Piotr Andruszkiewicz. 2019. Wut at semeval-2019 task 9: Domain-adversarial neural networks for domain adaptation in suggestion mining. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 1262–1266.

Douglas M Kline and Victor L Berardi. 2005. Revisiting squared-error and cross-entropy functions for training neural network classifiers. *Neural Computing & Applications*, 14(4):310–318.

Ludmila I Kuncheva, James C Bezdek, and Robert PW Duin. 2001. Decision templates for multiple classifier fusion: an experimental comparison. *Pattern recognition*, 34(2):299–314.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Mounica Maddela and Wei Xu. 2018. A word-complexity lexicon and a neural readability ranking model for lexical simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3749–3760.

Robert McHardy, Heike Adel, and Roman Klinger. 2019. Adversarial training for satire detection: Controlling for confounding variables. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 660–665.

Gustavo Paetzold and Lucia Specia. 2016a. Benchmarking lexical simplification systems. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3074–3080.

Gustavo Paetzold and Lucia Specia. 2016b. Semeval 2016 task 11: Complex word identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 560–569.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001.

Robi Polikar. 2006. Ensemble based systems in decision making. *IEEE Circuits and systems magazine*, 6(3):21–45.

Sebastian Schrom, Stephan Hasler, and Jürgen Adamy. 2021. Improved multi-source domain adaptation by preservation of factors. *Image and Vision Computing*, page 104209.

Matthew Shardlow, Richard Evans, Gustavo Paetzold, and Marcos Zampieri. 2021a. Semeval-2021 task 1: Lexical complexity prediction. In *Proceedings of the 14th International Workshop on Semantic Evaluation (SemEval-2021)*.

Matthew Shardlow, Richard Evans, and Marcos Zampieri. 2021b. Predicting lexical complexity in english texts. *arXiv preprint arXiv:2102.08773*.

Matthew Shardlow, Marcos Zampieri, and Michael Cooper. 2020. Complex—a new corpus for lexical complexity predicition from likertscale data. In *Proceedings of the 1st Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI)*, pages 57–62.

Yuhua Tang, Zhipeng Lin, Haotian Wang, and Liyang Xu. 2020. Adversarial mixup synthesis training for unsupervised domain adaptation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3727–3731. IEEE.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010.

Giorgos Vernikos, Katerina Margatina, Alexandra Chronopoulou, and Ion Androutsopoulos. 2020. Domain adversarial fine-tuning as an effective regularizer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 3103–3112.

Ke Wang, Gong Zhang, and Henry Leung. 2019. Sar target recognition based on cross-domain and cross-task transfer learning. *IEEE Access*, 7:153391–153399.

Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo Paetzold, Lucia Specia, Sanja Štajner, Anaïs Tack, and Marcos Zampieri. 2018. A report on the complex word identification shared task 2018. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 66–78.

George-Eduard Zaharia, Dumitru-Clementin Cercel, and Mihai Dascalu. 2020. Cross-lingual transfer learning for complex word identification. In *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 384–390. IEEE.

George-Eduard Zaharia, Dumitru-Clementin Cercel, and Mihai Dascalu. 2021. Upb at semeval-2021 task 1: Combining deep learning and hand-crafted features for lexical complexity prediction. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 609–616.

Dejiao Zhang, Ramesh Nallapati, Henghui Zhu, Feng Nan, Cicero dos Santos, Kathleen McKeown, and Bing Xiang. 2020. Unsupervised domain adaptation for cross-lingual text labeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 3527–3536.

Zhilu Zhang and Mert R Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 8792–8802.