EMBEDDIA

Cross-Lingual Embeddings for Less-Represented Languages in European News Media

Research and Innovation Action
Call: H2020-ICT-2018-1
Call topic: ICT-29-2018 A multilingual Next generation Internet
Project start: 1 January 2019 Project duration: 36 months

D3.6: Final cross-lingual comment filtering technology (T3.2)

Executive summary

This deliverable summarises the progress and outputs achieved on Task T3.2 of the EMBEDDIA project. Task T3.2 aims to develop cross-lingual tools for automatic filtering of user-generated comments in news media. We give a brief review of existing work in news comment filtering and related tasks, and initial EMBEDDIA progress already reported at month M18 in the previous deliverable for this task, D3.3. We then summarise our progress since then in developing new filtering classifiers directly for EMBEDDIA data and languages, in evaluating them on media partner datasets, and in applying the outputs of other work packages to develop advanced versions. The latest versions can use cross-lingual transfer to give good performance in less-resourced languages in zero-shot and few-shot settings, and can jointly infer topic information and use it in improved filtering accuracy. Finally, we outline some ongoing work using new transfer learning approaches that will lead to final results to be evaluated in D3.7.

Partner in charge: TEXTA

Project co-funded by the European Commission within Horizon 2020 Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	–
RE	Restricted to a group specified by the Consortium (including the Commission Services)	–
CO	Confidential, only for members of the Consortium (including the Commission Services)	–

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825153

Deliverable Information

Document administrative information	
Project acronym:	**EMBEDDIA**
Project number:	**825153**
Deliverable number:	**D3.6**
Deliverable full title:	**Final cross-lingual comment filtering technology**
Deliverable short title:	**Final cross-lingual comment filtering**
Document identifier:	**EMBEDDIA-D36-FinalCrosslingualCommentFiltering-T32-submitted**
Lead partner short name:	**TEXTA**
Report version:	**submitted**
Report submission date:	**31/10/2021**
Dissemination level:	**PU**
Nature:	**R = Report**
Lead author(s):	**Matthew Purver (QMUL), Ravi Shekhar (QMUL), Marit Asula (TEXTA)**
Co-author(s):	**Linda Freienthal (TEXTA), Andraž Pelicon (JSI), Senja Pollak (JSI), Marko Robnik-Šikonja (UL)**
Status:	**__ draft, __ final, _x_ submitted**

The EMBEDDIA Consortium partner responsible for this deliverable has addressed all comments received. Changes to this document are detailed in the change log table below.

Change log

Date	Version number	Author/Editor	Summary of changes made
01/09/2021	v0.1	Matthew Purver (QMUL)	Initial draft.
01/10/2021	v1.0	Matthew Purver (QMUL), Ravi Shekhar (QMUL)	First complete draft.
05/10/2021	v1.1	Marit Asula (TEXTA), Linda Freienthal (TEXTA)	End-user development details.
08/10/2021	v1.2	Matthew Purver (QMUL)	Submitted for internal review.
10/10/2021	v1.3	Hannu Toivonen (UH)	Internal review.
11/10/2021	v1.4	Adrian Cabrera (ULR)	Internal review.
13/10/2021	v1.5	Matthew Purver (QMUL), Marit Asula (TEXTA)	Revision based on internal reviews.
20/10/2021	prefinal	Nada Lavrač (JSI)	Report quality checked.
26/10/2021	final	Matthew Purver (QMUL)	Report finalized.
29/10/2021	submitted	Tina Anžič (JSI)	Report submitted.

Table of Contents

List of abbreviations

API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
CNN	Convolutional Neural Network
DNN	Deep Neural Network
DTD	Document-Topic Distribution
DTE	Document-Topic Embedding
ELMo	Embeddings from Language Models
ETM	Embedded Topic Model
IAC	Internet Argument Corpus
IRC	Internet Relay Chat
LASER	Language Agnostic SEntence Representations
LDA	Latent Dirichlet Allocation
LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
mBERT	Multilingual BERT
NER	Named Entity Recognition
NLP	Natural Language Processing
NN	Neural Network
NYT	New York Times
POS	Part Of Speech
RNN	Recurrent Neural Network
UGC	User-Generated Content
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine

1 Introduction

The EMBEDDIA project aims to improve cross-lingual transfer of language resources and trained models using word embeddings and cross-lingual technologies, with a focus on nine languages: Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene, and Swedish. Work package WP3 aims to apply EMBEDDIA's cross-lingual advances to help news media companies better serve their audience by understanding and analysing their reactions, and assuring the safety, fairness and integrity of their participation in public internet spaces. In Task T3.2, the focus is on automatic moderation and filtering of user-generated content (UGC), primarily the comments readers post under news articles.

The overall objective of workpackage WP3 is to apply EMBEDDIA's cross-lingual technologies to understand and analyse the reactions of multilingual news audiences. The specific objectives of WP3 are as follows:

- O3.1 Advance cross-lingual context and opinion analysis, via Task T3.1;
- O3.2 Develop cross-lingual comment filtering, via Task T3.2;
- O3.3 Develop techniques for report generation from multilingual comments, via Task T3.3.

The objective of this task T3.2 is therefore to develop cross-lingual methods for comment filtering. Work on user needs in WP6 Task T6.1 identified automatic comment filtering as a key requirement of media industry users: helping media partners deal with their need to quickly moderate large volumes of user-generated comments.

Our approach on Task T3.2 has been as follows. Our first steps were to develop classifiers for specific related tasks (sub-tasks of the overall filtering task, e.g. offensive language detection), using methods developed and tested in Task T3.1, together with already available and trusted social media datasets; these showed the viability of these methods, with good accuracy levels, as long as suitable resources exist (Pelicon et al., 2019; Miok et al., 2019). We then applied these to the more challenging data from real news comment moderation, provided by media partners, again showing viability (Shekhar et al., 2020), but showing that the noise inherent in such real-world datasets reduces performance, suggesting that methods for *intermediate* training (including other datasets in the training, ideally in other, better-resourced languages) could help boost performance.

Our recent focus, as reported in this deliverable, has therefore been on this step: the use of multi-lingual embedding models that can be first pre-trained in suitable languages; next, used to develop base classifiers via intermediate training on related datasets in available (usually well-resourced) languages; and then finally fine-tuned on the target language task and data, even if little data is available. Results using this approach are presented here. We show that the approach is viable, with cross-lingual training giving reasonable performance on comment filtering even with general pre-trained models (e.g. mBERT Devlin et al., 2019); and that this can be significantly improved by using EMBEDDIA cross-lingual BERT models from WP1 T1.2 (Ulčar & Robnik-Šikonja, 2020). These produce models with impressive performance, equalling the ideal accuracy that can be achieved with full target-language training data even when only 30% of the data is available, and with good performance even with no target-language training data. Classifiers based on these models have now been implemented and made public, and we have subsequently investigated a range of improvements.

The main contributions presented in this report (in the order of appearance) are as follows:

- Methods for cross-lingual training of classifiers for comment filtering, based on embeddings models from WP1, and equalling the accuracy of classifiers trained in the standard monolingual way, but with much less target-language training data (Pelicon, Shekhar, Škrlj, et al., 2021).
- Classifier models for comment filtering, including hate speech and offensive language detection, trained on real news comment data and in EMBEDDIA less-resourced languages (Pelicon, Shekhar, Martinc, et al., 2021; Pelicon, Shekhar, Škrlj, et al., 2021).

- Methods for improving the accuracy, confidence and interpretability of comment filtering classifiers by incorporating topic information (Zosa et al., 2021).
- Implemented multi-lingual classifier code and models, available with code for research use and behind APIs for software integration. A Dockerized version is available for integration with the EMBEDDIA Media Assistant in WP6.

This report is split into 6 further sections. Section 2 summarises related work and progress in the project so far in comment filtering. In Section 3, we describe our recent progress using cross-lingual methods to train effective classifiers with little data. Section 4 describes our progress in making these available outside the project and building practical tools. Section 5 then describes further research work into improving various aspects of the models. Section 6 summarises our conclusions and main findings, and outlines the connection to other ongoing EMBEDDIA tasks. Section 7 then summarises the main concrete outputs of this work, and the appendices include the papers on which the main content sections are based.

2 Background

This section explains the background to this work. First, we motivate it via the needs of the news media industry; next, we outline the research background, both the state of the art before the project began, and progress up to the most recent previous deliverable for this task, D3.3. The descriptions given here are brief; all sections have been covered in more detail in D3.3.

2.1 User needs

Work on user needs in WP6 Task T6.1 identified automatic comment filtering as a primary need for news media users, to help them quickly moderate large volumes of user-generated comments (see WP6, particularly deliverable D6.5). The primary requirements are summed up by the user stories given in deliverable D6.5, the relevant one repeated here for convenience as Figure 1. This describes the problem that must be solved, and the way in which an ideal future version of the EMBEDDIA tools would be used to do that.

Note that a moderator's blocking decisions, and therefore the output of an automatic filtering classifier, must take many phenomena into account, including spam, threatening language and misinformation. Two of the most important categories are generally offensive language and targeted hate speech, and it is these which many of our experiments have therefore focused on (see Section 3 below).

2.2 Work outside the EMBEDDIA project

Little previous work specifically on automatic moderation of news comments exists: Pavlopoulos et al. (2017b,a) address the problem in the Greek language, using a dataset of 1.6M comments with labels derived from the newspaper's human moderators and journalists; they test a range of neural network-based classifiers and achieve encouraging performance with AUC scores (area under the ROC curve) of 0.75-0.85 depending on the data subset. However, this work and the resulting models are specific to the Greek language, to the particular newspaper datset, and to the particular moderation policy of the newspaper studied (Gazzetta).

Other work with reader comments on news exists but addresses different tasks. Kolhatkar et al. (2019) and Napoles et al. (2017) investigate constructivity in comments; Barker et al. (2016) investigate quality of comments and their use in summarisation. Wulczyn et al. (2017) and Zhang et al. (2018) investigate detection of personal attacks and toxicity in user comments, but on Wikipedia articles rather than news. None directly address moderation or filtering, and all are limited to English.

Branko is a moderator at 24sata, the largest-circulation daily newspaper in Croatia. 24sata reaches about 2 million readers daily, and many of them post comments about online articles: on an average day, about 8000 comments come in, spread over several hundred articles. Unfortunately, many comments (usually between 5% and 10%) need to be blocked to prevent them appearing online: they might be offensive, dangerous, or legally compromising. This is Branko's job.

Until now, the task of comment filtering and moderation had to be performed almost entirely manually. This is time-consuming and skilled work: the newspaper has a complex moderation policy, as comments may need blocking for a variety of reasons. Some are irrelevant spam or advertising, some contain disinformation, some are threatening or hateful, some obscene or illegal, some written in foreign languages . . . so filtering through them all and making consistent decisions is difficult, especially at peak times when over 1,000 per hour may be coming in. Branko uses a system which flags comments that match a list of blacklisted keywords, but this isn't very accurate and is hard to keep up to date as new topics get discussed. With the current COVID-19 crisis, for example, new kinds of spam, fake stories and ethnically-targeted hate speech emerge very fast, and the word lists can't keep up. That means that Branko largely has to rely on fast reading and experience.

The new EMBEDDIA tools for automated comment moderation have made Branko's job much easier. Comments are filtered in real time, automatically detecting those which are most likely to need blocking, ranking them by severity, and labelling them as to which part of the 24sata policy they seem to break. The final decision is left to Branko, but now he can easily prioritise the worst cases first, and make sure they don't appear on the site, without having to read through all the others. He can then check less severe cases, and can leave unproblematic comments where the classifier is very confident for a less busy time. Branko's final decisions are stored and fed back to the system, so that it learns over time to improve, and to adapt to new vocabulary as new topics and stories develop.

Figure 1: User story from Deliverable D6.5: Comment filtering at 24sata, provided by Croatian EMBEDDIA partner Trikoder (Styria Group).

More resources are available, and more work has been done, on related specific tasks that correspond to particular phenomena or behaviours that moderators seek to block: in particular, work on the detection of offensive language and hate speech, mostly focusing on UGC in social media. Many public datasets have been created and distributed, many shared tasks have been run, and many classification systems developed and tested, although the exact definitions of the phenomena of interest vary with task and dataset – see Deliverable D3.1 for details.

Most work in this area is based on social media (mainly Twitter) posts, and is monolingual, mostly in English (Wulczyn et al., 2017; Davidson et al., 2017). Most shared tasks organised on the topic of hate or offensive speech have been English-only (e.g. OffensEval (Zampieri et al., 2019b)), although some more recent experiments and tasks have moved towards multilingual challenges (e.g. Basile et al., 2019; Ousidhoum et al., 2019; Zampieri et al., 2020). Performance varies widely with dataset, domain and language. However, these still generally focus on well-resourced languages: for example, Ousidhoum et al. (2019), use English, French and Arabic tweets; Zampieri et al. (2020) use English, Arabic, Danish, Greek, and Turkish. We are not aware of any data resources or tools in our primary target languages Croatian and Estonian.

2.3 Work within the project up to D3.3

Due to the lack of prior art applied to news comment data, and the lack of work in languages other than English, our overall approach in this task has been to incrementally move from known sub-problems (e.g. offensive language detection in English social media), to real news comment data, to less-resourced languages, and to the use of cross-lingual training.

Our first steps were on monolingual data, and started with existing, publicly available datasets in well-resourced languages; we developed monolingual classifiers for hate speech and abuse detection, with good accuracy on on standard datasets, including 4th place in the SemEval 2019 OffensEval task (Pelicon et al., 2019). We then moved on to apply these methods to the broader task of automatic news comment filtering, trained from real moderator behaviour, and in less-resourced languages, but still using a traditional monolingual train-and-test approach; this gave reasonable accuracy on EMBEDDIA news media partner comment data in EMBEDDIA project languages (Croatian, Estonian) (Shekhar et al., 2020).

The focus since that work has been on improving the ability to train classifiers and achieve good accuracy in the face of limited target-language data, by using cross-lingual training to leverage the information in existing annotated datasets in other, better-resourced languages. Initial work on this cross-lingual training looked at both standard hate speech datasets and EMBEDDIA news comment data, with training on available standard (e.g. English) datasets and transfer to EMBEDDIA project languages (Slovene, Croatian); it showed the viability of the approach in principle, but gave limited performance (Marinšek, 2019). Some improvements were shown possible by using the new cross-lingual BERT models from WP1 T1.2, but at the time of deliverable D3.3 this work was not yet completed (reported then as Pelicon et al., in preparation).

2.4 Summary and motivation

Our overall challenge in this task is to develop tools for automated news comment filtering in less-resourced languages, particularly in Croatian and Estonian, in the face of a lack of annotated data in those target languages. Success in this would provide general methods by which useful classification tools can be built without requiring significant manual annotation and development effort. Since D3.3, we have therefore focused our efforts on evaluating and improving our cross-lingual methods, developing classifiers that can be trained on related datasets in different (better-resourced) source languages and domains, and showing how good performance can be achieved in the target languages with minimal extra effort. Section 3 describes our final evaluations of our general cross-lingual methods; Section 4 describes the work we have done making the resulting tools available and testing them in more realistic end-user settings; and Section 5 then describes recent research into ways of improving their outputs.

3 Cross-lingual models for comment filtering

Most research in the area of offensive language and hate speech detection is still done in monolingual settings; for less-resourced language and for new domains, this is problematic, as suitable annotated datasets are unlikely to exist. Effective cross-lingual models would allow training initial models on related datasets in better-resourced languages, and using them either in a zero-shot (no target language training data at all) or few-shot (only small amounts of target-language training data) settings. In Deliverable D3.3 at M18 we included a preliminary study using standard multi-lingual pre-trained models (Marinšek, 2019) and initial as-yet unpublished work in improving results by using EMBEDDIA models specifically created in WP1 for the languages of interest here (Pelicon et al., in preparation). Since then, we have completed this work, developing and releasing models using both approaches; we describe these here.

3.1 Using standard pre-trained models

Our first cross-lingual models were based on existing, publicly available multilingual encoders; these models produce embedded representations of words and texts in which different languages are projected into a shared embedding space, thus potentially making them suitable for cross-lingual learn-

ing in which annotated datasets in a well-resourced source language can be used to learn classifier weights, which can then be applied to representations in the target language (even though little or no training data in that language may be used). We used two such multilingual models: Bidirectional Encoder Representations from Transformers (BERT, Devlin et al., 2019) and Language-Agnostic SEntence Representations (LASER, Artetxe & Schwenk, 2019).

In these experiments, we applied this approach to build classifiers for offensive language and hate speech, two major categories of language that should be blocked by moderators if appearing in comments, according to many newspaper policies (including those of our media partner 24sata, see (Shekhar et al., 2020) and Deliverable D3.1). Following standard approaches for classification, for BERT we attached a classification layer with a softmax activation function, and for LASER we used a multilayer perceptron classifier with RELU activation function.

We used existing public datasets, annotated for public shared tasks in hate speech and offensive language detection; all were based on Twitter data rather than news comments, but this might be expected to contain similar informal short-format text. For the offensive language detection task, we trained on the English (EN) training subset of the OLID dataset (Zampieri et al., 2019a), and evaluated on the corresponding English test set and on the test subset of the German (DE) GermEval 2018 dataset (Wiegand et al., 2018). For the hate speech detection task, we trained classifiers on the English training set from the HatEval dataset (Basile et al., 2019), and then tested on the English and Spanish (ES) test sets from the same HatEval competition, the German (DE) IGW hate speech dataset (Ross et al., 2016), an Indonesian (ID) hate speech dataset (Ibrohim & Budi, 2019) and the Arabic (AR) hate speech dataset LHSAB (Mulki et al., 2019).

Table 1: Results of the hate speech classification task (models trained on the English hatEval dataset) and offensive language classification task (models trained on the English OLID dataset) in comparison to the monolingual results as reported in the literature. The forward slash ('/') denotes results which are not reported in the literature. Figures marked with * denote results obtained on a different test split.

	Cross-lingual hate speech classification									
	Accuracy					F1-macro				
Model	EN	ES	DE	ID	AR	EN	ES	DE	ID	AR
LASER	**0.5241**	**0.6562**	0.5041	0.5755	**0.7013**	0.4994	0.6538	0.4630	**0.5172**	0.5500
BERT	0.5091	0.6313	**0.6369**	0.5823	0.6264	0.4341	0.5839	**0.6886**	0.4603	0.5033
Reported	/	/	/	**0.7353***	**0.9060***	**0.6510**	**0.7300**	/	/	**0.8930***
Majority	0.6000	0.6000	0.8500	0.5800	0.6200	0.3600	0.3700	0.4600	0.3700	0.3800
	Cross-lingual offensive language classification									
LASER	0.7500	/	0.7129	/	/	0.6823	/	0.6508	/	/
BERT	**0.8279**	/	**0.7148**	/	/	0.8263	/	0.7067	/	/
Reported	/	/	/	/	/	**0.829**	/	**0.7677**	/	/
Majority	0.6700	/	0.6600	/	/	0.4200	/	0.4000	/	/

Table 1 shows the results, with our cross-lingual classifiers' performance compared against the majority class baseline and against the results reported using monolingual methods in the literature. We can see that all models outperform the majority class baseline in terms of F1-score (accuracy figures for the hate speech task look worse, but are misleading due to the high class imbalance); and that BERT generally outperforms LASER. This is promising, in that it offers a practical approach for cross-lingual training where no target-language data is available, for the comment filtering task.

However, we can see that performance shows significant drops as compared to the monolingual state of the art: while the cross-lingual approach implemented here is robust enough to give some useful performance, it is significantly less than the monolingual equivalents which could be trained given available annotated target-language data.

In the next section, we show how this problem can be largely solved via the use of more specific pretrained language models, as developed in WP1.

This work is described in full in (Pelicon, Shekhar, Martinc, et al., 2021), attached here as Appendix A.

3.2 Using EMBEDDIA models

Section 3.1 shows that the general approach of cross-lingual training may be suitable for the comment filtering task, but fails to show competitive performance when using standard pre-trained models; and the experiments did not test the methods on EMBEDDIA project languages or on real news comment data. In this section, we address these issues, incorporating outputs of WP1, and show competitive cross-lingual performance in zero-shot and few-shot settings. This work builds on initial work reported in Deliverable D3.3, which was at that time in draft form; it has now been completed and fully evaluated, with an extended analysis of the model's behaviour, and published.

Here, we take the same general approach to cross-lingual training: training a classifier on an annotated dataset in some source language, and then applying to a test set in a different target language. However, here we use the new cross-lingual embedding models developed for the EMBEDDIA languages in WP1, rather than the standard multilingual BERT used in Section 3.1. By using a model based on more information about the target language, we expect improvements in terms of both overall classifier accuracy and transfer between languages. We examine its ability to support cross-lingual transfer, together with the effect of combining source-language training data with target-language training data in varying amounts. This simulates the progression from a *zero-shot* setting (in which no target-language training data is available) to *few-shot* settings (in which some target-language data is available), and evaluates the performance trade-offs.

Our datasets were taken from a range of different UGC domains and languages: standard datasets from shared tasks in English, Arabic and German, all taken from Twitter and labelled for offensive language (Zampieri et al., 2019b, 2020; Wiegand et al., 2018); a Slovenian language social media dataset from Facebook, labelled for offensive language (Ljubešić et al., 2019); and the EMBEDDIA 24sata news comment data (Shekhar et al., 2020; Pollak et al., 2021), labelled by 24sata's actual moderation process, within which we selected the subset moderated because it broke the 24sata policy against hate speech.

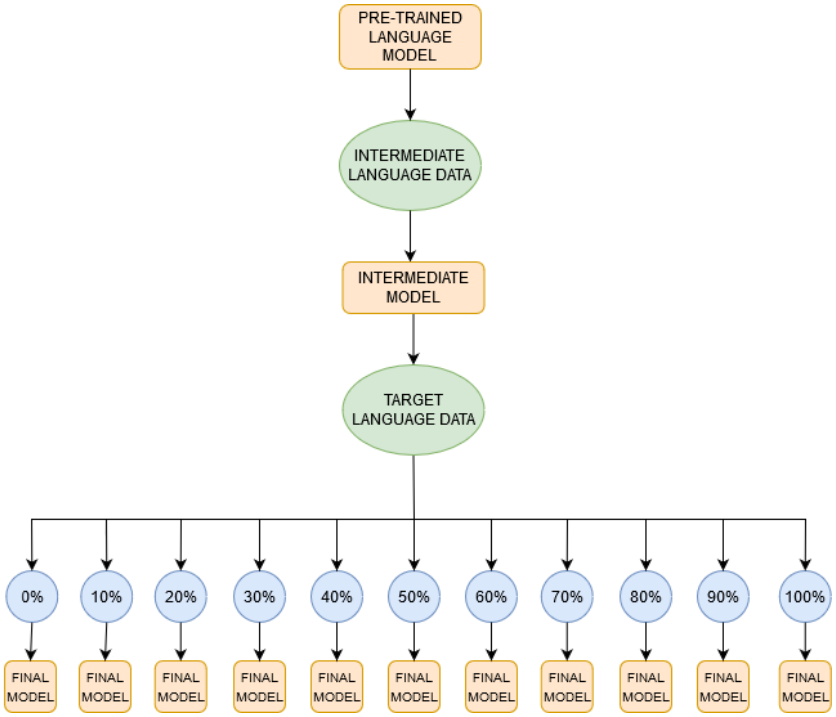
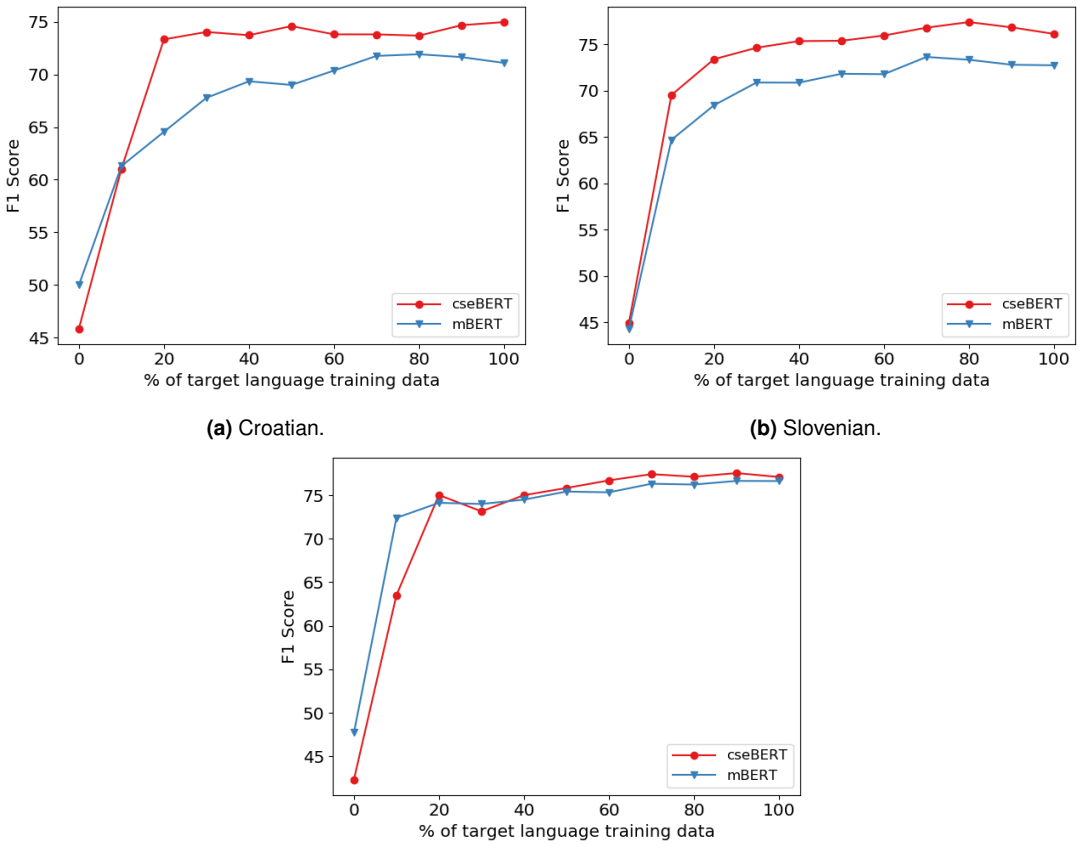


Figure 2: A schematic illustration of the training regime: we take a *pre-trained* language model; further train it on data in one or more *intermediate* non-target languages to produce an *intermediate* model; then fine-tune the result by progressively adding data in the *target* language to produce the *final* model tested.

As a multi-lingual embedding model we used the new EMBEDDIA CroSloEngual BERT (hereafter, cseBERT), a tri-lingual model for English, Croatian and Slovenian developed in WP1 (Ulčar & Robnik-Šikonja, 2020). Figure 2 shows the experimental setup, which allowed us to investigate multiple comparisons of the effects of varying amounts of source-language intermediate training, in varying combinations of languages, and of varying amounts of target-language training (with 0% corresponding to the zero-shot setting, and 100% to full dataset availability). Here, we present a few main conclusions.

Figure 3 shows the improvement gained by using the cseBERT model compared to the standard mBERT model in a monolingual setting (no cross-lingual intermediate training). For less-resourced languages Croatian and Slovenian, cseBERT gives improvements of 5-10% in overall macro-averaged F1-score, across the range of training dataset size, ending with around 5% improvement when all training data is available (100% on the x-axis). The effect is specific to the less-resourced languages that are underrepresented in mBERT: the English results show little to no improvement (Figure 3(c)).



(a) Croatian.

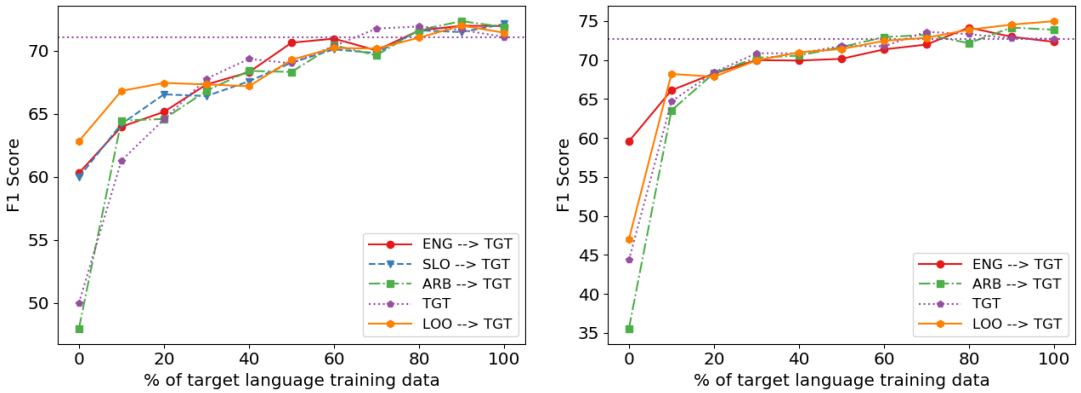
(b) Slovenian.

(c) English.

Figure 3: Effect of different pre-trained LMs (mBERT vs cseBERT), with varying amount of target language training data in the fine-tuning step, and no intermediate training.

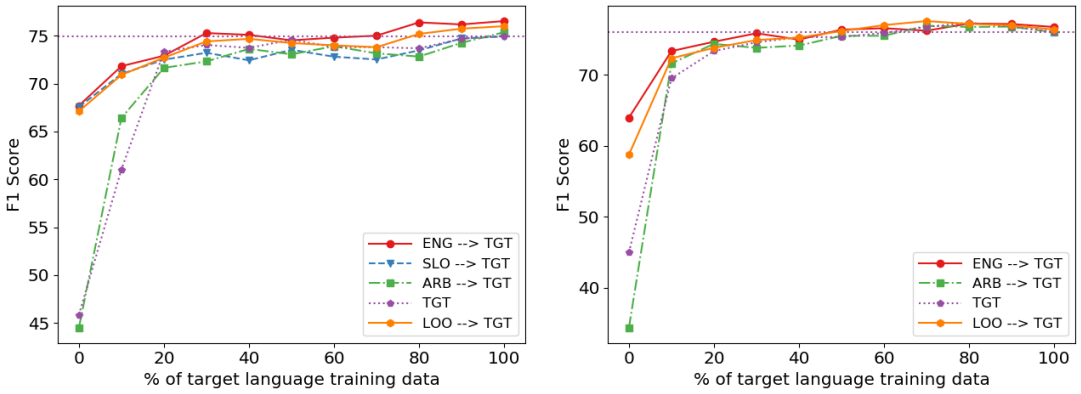
Figures 4 and 5 then show the cross-lingual equivalents, using mBERT (Figure 4) and cseBERT (Figure 5). The graphs show what can be achieved by cross-lingual intermediate training in selected individual languages (ENG, SLO and ARB), on all languages other than the target (LOO) and the comparison with the monolingual approach (TGT, i.e. no cross-lingual intermediate training). The dotted horizontal line shows the ideal performance achieved with the standard monolingual approach with 100% of training data.

As Figure 4 shows, using the standard mBERT does give some effective cross-lingual transfer, but that performance drops significantly in low-data settings. The zero-shot setting (0% on the x-axis) shows F1-



(a) Croatian. **(b)** Slovenian.

Figure 4: Effect of different intermediate training languages, with varying amount of target language training data in the fine-tuning step, using mBERT. TGT: Only fine-tuned on target language (no intermediate training). ENG/SLO/AR→TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO→TGT: Intermediate training on all non-target languages, then fine-tuning on target language.



(a) Croatian. **(b)** Slovenian.

Figure 5: Effect of different intermediate training language with varying amount of target training data, using cseBERT. TGT: Only fine-tuned on target language (no intermediate training). ENG/SLO/AR→TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO→TGT: Intermediate training on all non-target languages, then fine-tuning on target language.

score 8-12% below the ideal in even the best cases, and few-shot settings up to 30% of target-language data still show drops of 3-5%. We also see that cross-lingual training in a less-related language Arabic can be similar to or worse than using no intermediate training at all. However, Figure 5 shows that using cseBERT gives large improvements: in zero-shot settings, performance drops are similar, but absolute performances noticeably higher; and in few-shot settings, the drops are less, with the ideal 100%-data performance reached by the time only 30% of target-language data is used. The cross-lingual training gives large boosts compared to the monolingual target-language only case (the TGT lines).

This work is described in full in (Pelicon, Shekhar, Škrlj, et al., 2021), attached here as Appendix B.

4 Developing practical tools

Section 3 shows that comment filtering classifiers can be created using our cross-lingual methodology, and that they can have good performance even with little target-language training data (comparable to the ideal case of monolingual training with full datasets) if suitable pre-trained language models are used. In this section, we describe progress towards making these models available for use outside the project, both as general research tools and as tools for use in industry.

4.1 Developing practical research tools

Our first step has been to make our models and methods generally available to the research community: we have deposited a range of our pre-trained classifier models, together with scripts for training and replicating the experiments of Section 3 above, in public repositories. The code and models are available for researchers, and have already been used by teams outside the EMBEDDIA projects for comparative experiments (see Korencic et al., 2021). The models have been implemented with an API front-end, for connection to other software components, and distributed as Docker images for easy installation and use (see Section 7 for URLs).

This work is described in (Pelicon, Shekhar, Martinc, et al., 2021), attached here as Appendix A.

4.2 Developing practical end-user tools

The second part of this effort has focused on applying the methods developed in the research so far directly to an end-user's task: assisting human moderators at Ekspress Meedia (ExM) by predicting comments that have a high probability that they should be blocked. To test the success of our methods in a realistic setting, we worked with ExM to gather and annotate new data, and then train a filtering model, based on the EMBEDDIA WP1 pre-trained embeddings, and test its accuracy with small amounts of training data.

4.2.1 Constructing a new end-user dataset

To provide both new evaluation data and as-yet unseen training data, we gathered a new dataset of news comments, annotating it for comments that should be blocked, and focussing particularly on toxic comments containing hate speech. By gathering new data, we avoid biasing our tests towards the data already gathered in previous years, and provided by ExM to the project during the development so far: the new data comes from a new time period, with new topics etc.

As raw comment volumes can be high, we took a filtered subset of the general ExM comment stream; to ensure that the dataset covered as much variety as possible in terms of linguistic phenomena and likely classifier performance (and therefore provide a thorough evaluation), we filtered on the basis not only of the labels provided by ExM's moderators, but the predictions given by two existing classifier models created in previous work. Although it might seemingly make sense to use only to the moderators' decisions (they should best mirror ExM's comment filtering standards), manual investigation revealed that the moderators' labelling tended to be rather inconsistent: comments with exactly the same text were sometimes blocked by the moderators and at other times not. Furthermore, comments that seemed to contain rather innocent content were often blocked, while comments that were clearly toxic were not. We therefore used three sources of information to filter the dataset:

- Model 1 (higher positive classification threshold compared to Model 2) predictions;
- Model 2 (lower positive classification threshold compared to Model 1) predictions;
- Moderator feedback.

Both Model 1 and Model 2 were Logistic Regression models, trained on a dataset automatically labeled based on manually constructed hate-speech lexicons.

To ensure the desired variety, we then constructed 10 subsets of the comments with various combinations of Model 1, Model 2 and the moderators' labels, ad shown in Table 3. Note that the common approach of using only comments that all three parties agreed on is not appropriate here: our goal is to create a subset which represents a range of not only comments that should definitely be blocked/kept, but those that are less clear, and likely to confuse classifiers. The final dataset consisted of **25,000 comments**.

This dataset was then annotated by two annotators separately hired for the task, and given a moderation tutorial by ExM. Again, annotator decisions often did not agree with each other. The label distribution of the annotated dataset can be seen in Table 4.

✗	Block
✓	Keep
	No Information

Table 2: Legend for Table 3

Model 1	✗	✓	✗	✗	✓	✗	✓	✗	✓	✓
Model 2	✗	✓	✗	✓	✗	✓	✗	✗	✓	✓
Moderator	✗	✓	✓	✓	✓	✗	✗			✗
Sample size	5000	5000	2000	1500	1500	2000	2000	2000	2000	2000

Table 3: The class distributions of the hate speech dataset passed to the annotators. The dataset was constructed by merging 10 subsets with different filters to guarantee as much variety as possible.

		Annotator 2			
		Block	**Keep**	**Undecided**	
Annotator 1	**Block**	783	750	83	**1616**
	Keep	1822	19937	40	**21799**
	Undecided	158	1420	7	**1585**
		2763	**22107**	**130**	**25000**

Table 4: Class distributions for the annotated dataset.

4.2.2 Developing a new comment filtering model for Estonian

This new annotated dataset was then used to develop a new comment moderation classifier for Estonian. We first constructed a balanced test set DS5 containing examples with clear annotation decisions: 500 comments randomly selected from the 783 examples that both annotators decided should be blocked, and 500 randomly selected from the 19,937 that both annotators decided should be kept. We then constructed a series of different training sets of increasing sizes and with different criteria used to generate the labels. In dataset DS1, labels were derived from decisions where both annotators agreed: this therefore contained the remaining 283 "Block" comments and a randomly selected 283 "Keep" comments. In dataset DS2, we added more data by using "Block" comments where only one of the annotators had labelled them as such, expanding the size to 3,096 examples of each class. In dataset DS3, the annotators' decisions were entirely ignored and the "Block" and "Keep" labels were taken from decisions that were unanimous between Model 1, Model 2 and the original ExM moderator(s). The test set DS5 had no overlap with any training set DS1-4.

We then used these training sets to train a range of BERT-based classifier models using the same methodology as in the sections above (using BERT's CLS token representation as a sentence embedding and passing through a linear classifier layer), and validated the results on the test set DS5. We

Table 5: Training and test dataset sizes and class distributions.

Symbol	Type	Requirements for "Block" label	# "Block"	# "Keep"
DS1	Train set	annotator 1 & annotator 2	283	283
DS2	Train set	annotator 1 || annotator 2	3096	3096
DS3	Train set	model 1 & model 2 & moderator	4799	4799
DS4	Train set	annotator 1 || annotator 2	3096	5263
DS5	Test set	annotator 1 & annotator 2	500	500

compared the EMBEDDIA WP1 trilingual RoBERTa model for English, Finnish and Estonian (see Deliverable D1.10) with an existing Estonian language BERT model, EstBERT (Tanvir et al., 2021). The first comparisons showed that results of models trained on DS2 were significantly better that the others (F1 score 0.85 vs. 0.62 and 0.37); we therefore selected DS2 for further experiments. We then trained a number of BERT models with various parameter combinations seen in Table 6 (IDs b1-b7). For comparison, we also trained logistic regression models with parameters displayed in Table 7. The EMBEDDIA EstRoBERTa outperformed EstBERT; the logistic regression models performed significantly worse with the highest F1 score of 0.73.

Although the resulting F1 scores were quite impressive, with the highest one being 0.86, the models tended to have higher recall and lower precision. This does not suit ExM's desired use case: ExM wanted to use the model for automatically deleting comments without additional moderation, meaning that the deployable model should have as high precision as possible. To increase precision, we therefore added a further 2168 "Keep" comments to the train set, resulting in class sizes 3096 for "Block" and 5263 for class "Keep" (dataset DS4 in Table 5). Using this to train a new model (model b8 in Table 6) improved precision from 0.8 to 0.9, while still maintaining decent recall (0.89) and F1 scores (0.89). This therefore gives a model with potential for deployment by the end user, created using a training set requiring only a few thousand annotator decisions.

Table 6: Parameters of BERT comment filtering models for Estonian.

ID	Dataset	BERT Model	Max Length	LR	Epochs	"Block" Ratio
b1	DS1	EMBEDDIA/est-roberta	128	0.0002	2	0.5
b2	DS2	EMBEDDIA/est-roberta	128	0.0002	2	0.5
b3	DS3	EMBEDDIA/est-roberta	128	0.0002	2	0.5
b4	DS2	tartuNLP/EstBERT	128	0.0002	2	0.5
b5	DS2	EMBEDDIA/est-roberta	200	0.00002	3	0.5
b6	DS2	tartuNLP/EstBERT	200	0.00002	3	0.5
b7	DS2	EMBEDDIA/est-roberta	300	0.00002	3	0.5
b8	DS4	EMBEDDIA/est-roberta	300	0.0002	2	0.37

Table 7: Parameters of Logistic Regression comment filtering models for Estonian.

ID	Dataset	Vectorizer	Input Type	"Block" Ratio
lr1	DS2	TF-IDF	stems	0.5
lr2	DS2	TF-IDF	lemmas	0.5
lr3	DS4	TF-IDF	stems	0.37

To showcase the model's ability to distinguish triggers based on context, we analyzed a potential trigger word "kuul" ("on the moon", "bullet" or "cool") in 4 different contexts. As shown in Table 9, the model was able to take the different contexts into account, and make correct predictions in all cases (an improvement over the keyword-based models used recently in many industry settings).

Table 8: Results of the Estonian comment filtering experiments. Figures are F1-scores, precision and recall for the positive "Block" class.

	b1	b2	b3	b4	b5	b6	b7	b8	lr1	lr2	lr3
F1-score	0.62	0.85	0.37	0.80	0.87	0.82	0.86	**0.89**	0.73	0.69	0.67
Precision	0.80	0.79	0.45	0.73	0.84	0.81	0.80	**0.90**	0.71	0.73	0.84
Recall	0.51	0.91	0.32	0.89	0.91	0.84	**0.94**	0.89	0.76	0.65	0.56

Table 9: Examples of comments containing the context sensitive trigger word “kuul”.

Comment (original ET)	Comment (EN translation)	True Label	Predicted Label (b8)
kuul maandus esimene inimene	the first man landed on the moon ("bullet" and "on the moon" both translate to "kuul" in Estonian)	keep	keep
kuul pähe kõikidele venelastele!	all Russians should get a bullet to the head!	block	block
kuul pähe kõikidele, kellele valitsus ei meeldi? See on politseiriik!	a bullet to the head to everyone who complains about the government? this is police state!	keep	keep
kuul! Eesti sportalsed hoiavat taset!	cool! Estonian athletes are still good!	keep	keep

5 Improving accuracy, interpretability and efficiency

Our most recent work has now turned to examining a range of approaches that might improve filtering performance, improve the utility or interpretability of its outputs, and achieve performance with less computational expense.

5.1 Integrating topic modelling

One direction investigated is the incorporation of information from the topic modelling work carried out in WP4 on news articles, and then adapted and applied to user comments in Task T3.1, as described in Deliverable D3.4. Our motivation is that the linguistic features associated with particular kinds of comment that require moderation might vary depending on their topical context: comments in different sections of a newspaper, and/or discussing different topics, might use language in different ways. A classifier that can incorporate this knowledge might therefore make better moderation decisions.

As shown in Task T3.1, this does indeed appear to be the case. Using the Embedded Topic Model (ETM, Dieng et al., 2020), we learned a topic model in which the topics are embedded in the same space as the (pre-trained) word embeddings, trained on one of our EMBEDDIA media partner news comment datasets, the 24sata comment dataset of c.21M comments on 476K articles from the years 2007-2019, written in Croatian (Shekhar et al., 2020). Examining the distribution of the inferred topics with comment moderation labels (taken directly from 24sata’s human moderator decisions), we see that different topics show different associations with the likelihood that a human moderator would block a comment, with these associations varying across different sections of the newspaper. Figure 6 shows an example for two 24sata news sections (*Lifestyle* and *Politics*), across blocked and non-blocked comments. Different topics can be seen to appear in different areas; some associations may be unintuitive, e.g. the association of the “Football cards” topic with blocking in the *Lifestyle* section, but turn out to be meaningful: commenters often discuss moderator’s blocking decisions as “yellow cards” or “red cards”, and this discussion is associated with further blocking decisions. The same association does not hold, of course, in the *Sports* section.

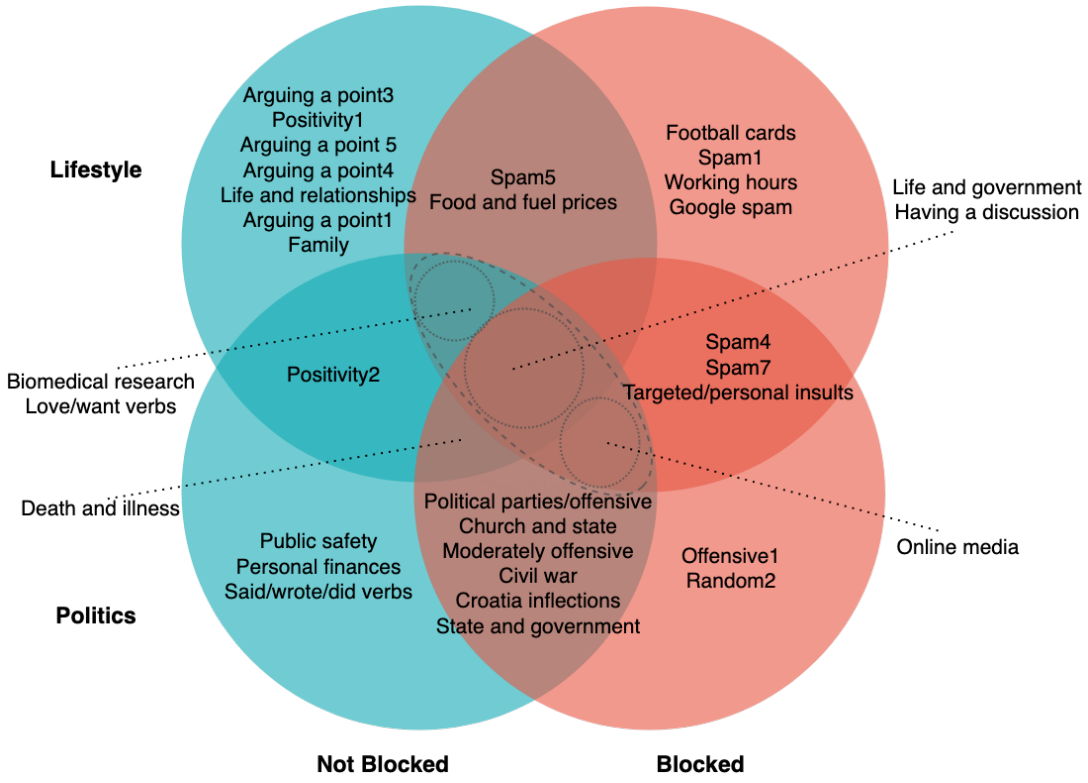


Figure 6: Top topics of the blocked and non-blocked comments for the entire test set (Zosa et al., 2021).

We therefore expect that this topic information will help improve the accuracy of comment filtering classifiers. We represent topic information in two ways: the **document-topic distribution (DTD)** of a text, and the **document-topic embedding (DTE)** (the weighted sum of the embeddings of the topics in a text, where the weight corresponds to the probability of the topic in that text). We tested two alternative models for fusing this information with the comment text, similar to the fusion architecture introduced in our earlier work on context analysis (see Rohanian et al., 2019, and the earlier Deliverable D3.2) – see Figure 7.

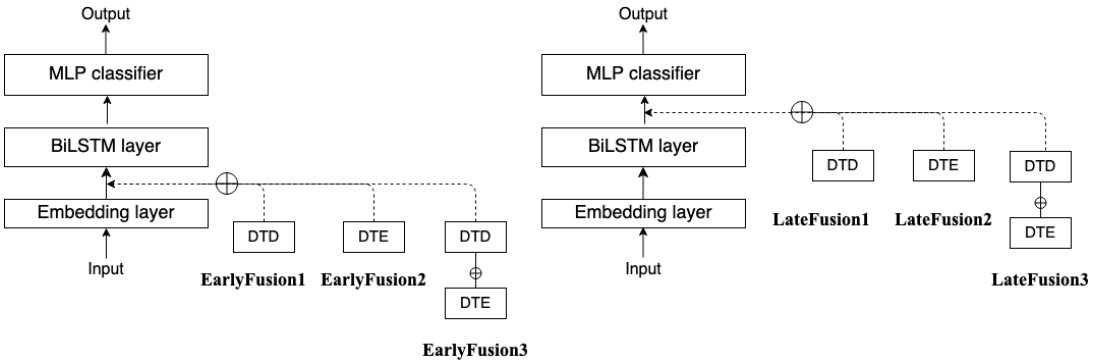


Figure 7: Network structure for topic/comment fusion (Zosa et al., 2021).

The resulting filtering performance is shown in Table 10: the use of topics in our model significantly improves classifier performance, giving boosts of 3-5% absolute improvement in F1-score over the standard text-only model. The model also gives more confident outputs (higher classifier confidences in cases where prediction is correct), which promises to be useful in a situation where high-confidence outputs are presented to a human user (as intended in end-user evaluation work to come in Task T3.4

and/or WP6). The model also provides topic distributions, interpretable as keywords, as a form of an explanation of its prediction.

Table 10: Classifier performance measured as macro-F1, comparing performance with and without topic fusion, for different 24sata news sections. Model variants EF=Early Fusion, LF=Late Fusion – see Figure 7.

Section – Subsection	Text only	Topics only			Text+Topic Combinations					
		DTD	DTE	DTD+E	EF1	EF2	EF3	LF1	LF2	LF3
All	62.97	62.20	59.3	58.33	66.33	66.58	65.61	**67.37**	66.22	66.95
Kolumne	59.86	59.65	56.25	55.33	62.40	62.90	63.13	63.25	62.38	**63.6**
Lifestyle	69.21	70.07	65.93	64.47	72.73	70.9	69.36	72.00	72.39	**72.92**
Show	61.97	61.30	58.62	57.60	65.24	65.63	64.26	**66.50**	65.00	65.86
Sport	63.22	61.42	58.61	57.90	67.11	67.86	66.74	**68.26**	67.14	67.82
Tech	64.87	66.37	63.17	62.55	67.72	68.74	67.65	68.76	67.68	**69.15**
Vijesti (*News*)	62.38	61.49	58.79	57.77	65.58	65.99	65.24	**66.77**	65.53	66.24
– Crna kronika	64.67	63.98	61.03	59.84	68.10	68.88	68.11	**69.60**	67.89	68.88
– Hrvatska	63.61	63.50	60.10	58.93	67.24	66.86	65.95	67.90	67.12	**67.95**
– Politika	57.93	56.49	54.95	54.20	60.51	61.52	60.84	**61.61**	60.63	61.30
– Svijet	63.58	62.55	59.62	58.35	66.83	66.95	66.33	68.44	67.21	67.57

This work is described in full in (Zosa et al., 2021), attached here as Appendix C.

5.2 Improving cross-lingual training

A second direction we have investigated is to improve the filtering classifier by improving the ability of the model to adapt its embeddings to the language and domain.

Most language models (LMs) are trained on broad, heterogeneous sources of data Devlin et al. (2019). These models work effectively on a range of downstream tasks; however, due to the unavailability of the specific domain knowledge, their performance is often less good than specific models for some domains such as biomedicine (Lee et al., 2020). One way to incorporate the required domain knowledge is via *domain adaptation* of the LM: the LM is further fine-tuned on domain-specific data to improve the performance on the downstream domain tasks (Gururangan et al., 2020).

This has similarities to the problem of applying large, multilingual LMs to less-resourced languages, as examined in WP1: multilingual LMs often perform poorly on less-resourced languages as they are under-represented in their training data and tokenizer vocabulary, and better performance can be obtained by using a more specific LM trained for the target languages (see results in WP1 Deliverable D1.10, and the results in Section 3.2 above). The idea of domain adaptation, though, takes a slightly different approach: fine-tuning a general LM, rather than training a specific LM from scratch.

In this work, then, we are interested in investigating the effect of domain fine-tuning on a comment filtering classifier, both (a) when a language-specific pre-trained model is available (for example, cseBERT (Ulčar & Robnik-Šikonja, 2020)), and (b) when it is not. Improvements in case (a) would be an overall advantage, giving generally improved performance over our best results so far; improvements in case (b) might help gain good performance without the computationally expensive task of specific target-language LM pre-training. As an exploratory study, we fine-tuned the general multilingual mBERT and the specific target-language cseBERT on the Croatian 24sata newspaper comments, and tested the effect of domain adaptation on our comment moderation task. For the domain LM finetuning, we used the 24sata comment data from 2007-2017, and for the comment moderation task, we used data from 2019 data.

Figure 8 shows our preliminary results: we can observe that domain fine-tuning improves the performance for both mBERT and cseBERT. However, the performance gained by mBERT is comparatively more than for cseBERT. Interestingly, the domain fine-tuned mBERT seems to achieve similar perfor-

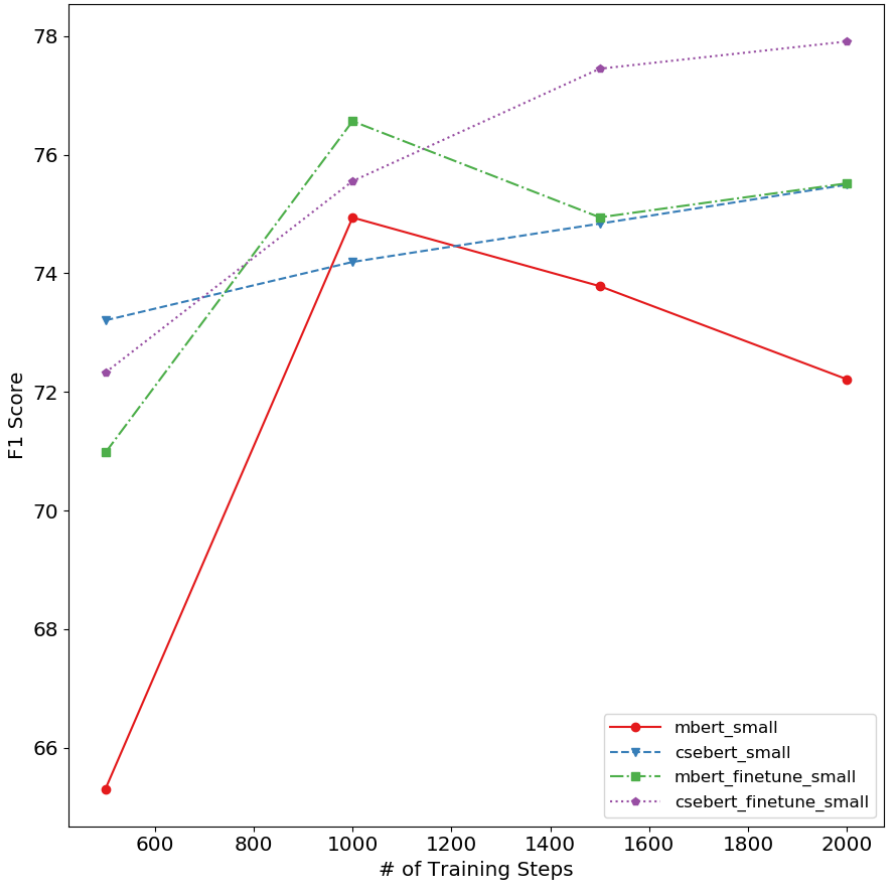


Figure 8: Effect of domain finetuning of LM.

mance as cseBERT without fine-tuning, perhaps offering a cheaper and easier alternative to full target-language pre-training. However, we can see that overall, having a language-specific LM is better than the generic multi-lingual LM; and that domain-specific fine-tuning further improves performance. As future work, we plan to test two hypotheses: first, the effect of generic vs. domain LM fine-tuning and then, the impact of adding domain vocabulary to improve LM fine-tuning.

6 Conclusions and further work

The objective of this task was to develop effective cross-lingual technologies for news UGC, i.e. news comment filtering. As Section 3 shows, we have succeeded in developing classifiers for filtering news comments based on the presence of offensive language, that can achieve good performance in less-resourced languages and on real news comment data, and that can achieve the same level of performance as a fully target-language trained tool by using cross-lingual training, together with fine-tuning on only ~30% of the amount of target-language data.

Section 4 showed that these tools can be applied to new end-user data, and trained to give high levels of accuracy, tuned to end user precision/recall requirements, with small amounts of annotated data. Section 5 described the recent research into possible improvements in terms of outputs and training approaches. Section 7 below gives details of the tools and code developed.

Next steps will extend our end-user testing into another project language (Croatian) and evaluate the use of the classifier outputs by moderators, as part of Task T3.4, to be reported in Deliverable D3.7.

7 Associated outputs

The work described in this deliverable has resulted in the following resources:

Description	URL	Availability
Code and models for comment filtering (Pelicon, Shekhar, Martinc, et al., 2021)	github.com/EMBEDDIA/hackashop2021_comment_filtering	Public (CC0)
Code for cross-lingual training (Pelicon, Shekhar, Škrlj, et al., 2021)	github.com/EMBEDDIA/cross-lingual_training_for_offensive_language_detection	Public (MIT)
Dockerized API for comment filtering (Pelicon, Shekhar, Škrlj, et al., 2021)	github.com/EMBEDDIA/comment-filter	Public (MIT)
Code for topic-based comment filtering (Zosa et al., 2021)	github.com/EMBEDDIA/croatian_topic_api	Public (MIT)
TEXTA Toolkit including Estonian comment filtering	github.com/EMBEDDIA/texta-rest	Public (GPL)

Parts of this work are also described in detail in the following publications, which are attached to this deliverable as appendices:

Citation	Status	Appendix
Pelicon, A., Shekhar, R., Martinc, M., Škrlj, B., Purver, M. & Pollak, S. (2021a). Zero-shot Cross-lingual Content Filtering: Offensive Language and Hate Speech Detection. In Proceedings of the EACL workshop on News Media Content Analysis and Automated Report Generation.	Published	Appendix A
Pelicon, A., Shekhar, R., Škrlj, B., Purver M. & Pollak, S. (2021b). Investigating Cross-Lingual Training for Offensive Language Detection. PeerJ Computer Science 7:e559.	Published	Appendix B
Zosa, E., Shekhar, R., Karan, M., & Purver, M. (2021). Not all comments are equal: Insights into comment moderation from a topic-aware model. In Proceedings of Recent Advances in Natural Language Processing (RANLP).	Published	Appendix C

Bibliography

Artetxe, M., & Schwenk, H. (2019). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, *7*, 597–610.

Barker, E., Paramita, M. L., Aker, A., Kurtic, E., Hepple, M., & Gaizauskas, R. (2016, September). The SENSEI annotated corpus: Human summaries of reader comment conversations in on-line news. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue* (pp. 42–52). Los Angeles: Association for Computational Linguistics.

Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Rangel Pardo, F. M., ... Sanguinetti, M. (2019, June). SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proc. SemEval* (pp. 54–63). Minneapolis, Minnesota, USA: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/S19-2007 doi: 10.18653/v1/S19-2007

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Eleventh International AAAI Conference on Web and Social Media.*

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (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)* (pp. 4171–4186).

Dieng, A. B., Ruiz, F. J. R., & Blei, D. M. (2020). Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, *8*, 439–453. Retrieved from https://www.aclweb.org/anthology/2020.tacl-1.29 doi: 10.1162/tacl_a_00325

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., & Smith, N. A. (2020, July). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 8342–8360). Online: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2020.acl-main.740 doi: 10.18653/v1/2020.acl-main.740

Ibrohim, M. O., & Budi, I. (2019, August). Multi-label hate speech and abusive language detection in Indonesian Twitter. In *Proc. 3rd Workshop on Abusive Language Online* (pp. 46–57). Florence, Italy: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W19-3506 doi: 10.18653/v1/W19-3506

Kolhatkar, V., Wu, H., Cavasso, L., Francis, E., Shukla, K., & Taboada, M. (2019, Nov 02). The SFU opinion and comments corpus: A corpus for the analysis of online news comments. *Corpus Pragmatics*.

Korencic, D., Baris, I., Fernandez, E., Leuschel, K., & Sánchez Salido, E. (2021, April). To block or not to block: Experiments with machine learning for news comment moderation. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation* (pp. 127–

133). Online: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2021.hackashop-1.18

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, *36*(4), 1234–1240.

Ljubešić, N., Fišer, D., & Erjavec, T. (2019). The FRENK Datasets of Socially Unacceptable Discourse in Slovene and English. In *International Conference on Text, Speech, and Dialogue* (pp. 103–114).

Marinšek, R. (2019). *Cross-lingual embeddings for hate speech detection in comments* (Unpublished master's thesis). University of Ljubljana, Faculty of Computer and Information Science.

Miok, K., Nguyen-Doan, D., Škrlj, B., Zaharie, D., & Robnik-Šikonja, M. (2019). Prediction uncertainty estimation for hate speech classification. In *International conference on statistical language and speech processing* (pp. 286–298). Springer.

Mulki, H., Haddad, H., Bechikh Ali, C., & Alshabani, H. (2019, August). L-HSAB: A Levantine Twitter dataset for hate speech and abusive language. In *Proc. 3rd Workshop on Abusive Language Online* (pp. 111–118). Florence, Italy: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W19-3512 doi: 10.18653/v1/W19-3512

Napoles, C., Tetreault, J., Rosata, E., Provenzale, B., & Pappu, A. (2017, April). Finding good conversations online: The Yahoo news annotated comments corpus. In *Proceedings of The 11th Linguistic Annotation Workshop* (pp. 13–23). Valencia, Spain: Association for Computational Linguistics.

Ousidhoum, N., Lin, Z., Zhang, H., Song, Y., & Yeung, D.-Y. (2019, November). Multilingual and multi-aspect hate speech analysis. 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)* (pp. 4675–4684). Hong Kong, China: Association for Computational Linguistics. Retrieved from https://aclanthology.org/D19-1474 doi: 10.18653/v1/D19-1474

Pavlopoulos, J., Malakasiotis, P., & Androutsopoulos, I. (2017a, September). Deeper attention to abusive user content moderation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 1125–1135). Copenhagen, Denmark: Association for Computational Linguistics.

Pavlopoulos, J., Malakasiotis, P., & Androutsopoulos, I. (2017b, August). Deep learning for user comment moderation. In *Proceedings of the First Workshop on Abusive Language Online* (pp. 25–35). Vancouver, BC, Canada: Association for Computational Linguistics.

Pelicon, A., Martinc, M., & Kralj Novak, P. (2019, June). Embeddia at SemEval-2019 task 6: Detecting hate with neural network and transfer learning approaches. In *Proceedings of the 13th International Workshop on Semantic Evaluation* (pp. 604–610). Minneapolis, Minnesota, USA: Association for Computational Linguistics.

Pelicon, A., Shekhar, R., Martinc, M., Škrlj, B., Purver, M., & Pollak, S. (2021, April). Zero-shot cross-lingual content filtering: Offensive language and hate speech detection. In H. Toivonen & M. Boggia (Eds.), *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation* (p. 30-34). Retrieved from http://www.eecs.qmul.ac.uk/~mpurver/papers/pelicon-et-al21eacl.pdf

Pelicon, A., Shekhar, R., Škrlj, B., Pollak, S., & Purver, M. (in preparation). *Zero-shot cross-lingual content filtering: Offensive language and hate speech detection.* (Draft)

Pelicon, A., Shekhar, R., Škrlj, B., Purver, M., & Pollak, S. (2021). Investigating cross-lingual training for offensive language detection. *PeerJ Computer Science*, *7*, e559. Retrieved from https://doi.org/10.7717/peerj-cs.559 doi: 10.7717/peerj-cs.559

Pollak, S., Robnik Šikonja, M., Purver, M., Boggia, M., Shekhar, R., Pranjić, M., ... Doucet, A. (2021, April). EMBEDDIA tools, datasets and challenges: Resources and hackathon contributions. In *Pro-*

ceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation. Association for Computational Linguistics.

Rohanian, M., Hough, J., & Purver, M. (2019). Detecting depression with word-level multimodal fusion. In *Proceedings of INTERSPEECH* (pp. 1443–1447). Graz, Austria: ISCA. (ISSN 1990-9772)

Ross, B., Rist, M., Carbonell, G., Cabrera, B., Kurowsky, N., & Wojatzki, M. (2016, sep). Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis. In M. Beißwenger, M. Wojatzki, & T. Zesch (Eds.), *Proc. NLP4CMC III: 3rd Workshop on Natural Language Processing for Computer-Mediated Communication* (Vol. 17, p. 6-9). Bochum.

Shekhar, R., Pranjić, M., Pollak, S., Pelicon, A., & Purver, M. (2020). Automating news comment moderation with limited resources: Benchmarking in Croatian and Estonian. *Journal of Language Technology and Computational Linguistics*, *34*, 49-79. (Special Issue on Offensive Language)

Tanvir, H., Kittask, C., Eiche, S., & Sirts, K. (2021, May 31–2 June). EstBERT: A pretrained language-specific BERT for Estonian. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)* (pp. 11–19). Reykjavik, Iceland (Online): Linköping University Electronic Press, Sweden. Retrieved from `https://aclanthology.org/2021.nodalida-main.2`

Ulčar, M., & Robnik-Šikonja, M. (2020). FinEst BERT and CroSloEngual BERT: less is more in multilingual models. In *Proceedings of Text, Speech, and Dialogue, TSD 2020.* doi: 10.1007/978-3-030-58323-1_11

Wiegand, M., Siegel, M., & Ruppenhofer, J. (2018). *Overview of the GermEval 2018 shared task on the identification of offensive language.*

Wulczyn, E., Thain, N., & Dixon, L. (2017). Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web* (pp. 1391–1399).

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., & Kumar, R. (2019a, June). Predicting the type and target of offensive posts in social media. In *Proc. NAACL-HLT* (pp. 1415–1420). Minneapolis, Minnesota: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/N19-1144` doi: 10.18653/v1/N19-1144

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., & Kumar, R. (2019b, June). SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation* (pp. 75–86). Minneapolis, Minnesota, USA: Association for Computational Linguistics.

Zampieri, M., Nakov, P., Rosenthal, S., Atanasova, P., Karadzhov, G., Mubarak, H., ... Çöltekin, Ç. (2020, December). SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation* (pp. 1425–1447). Barcelona (online): International Committee for Computational Linguistics. Retrieved from `https://aclanthology.org/2020.semeval-1.188`

Zhang, J., Chang, J., Danescu-Niculescu-Mizil, C., Dixon, L., Hua, Y., Taraborelli, D., & Thain, N. (2018, July). Conversations gone awry: Detecting early signs of conversational failure. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1350–1361). Melbourne, Australia: Association for Computational Linguistics.

Zosa, E., Shekhar, R., Karan, M., & Purver, M. (2021, September). Not all comments are equal: Insights into comment moderation from a topic-aware model. In G. Angelova, M. Kunilovskaya, R. Mitkov, & I. Nikolova-Koleva (Eds.), *Proceedings of Recent Advances in Natural Language Processing (RANLP)* (pp. 1652–1662). online. Retrieved from `https://doi.org/10.26615/978-954-452-072-4_185` doi: 10.26615/978-954-452-072-4_185

Appendix A: Zero-shot Cross-lingual Content Filtering: Offensive Language and Hate Speech Detection

Zero-shot Cross-lingual Content Filtering: Offensive Language and Hate Speech Detection

Andraž Pelicon[1], **Ravi Shekhar**[3], **Matej Martinc**[1,2]
Blaž Škrlj[1,2], **Matthew Purver**[2,3], **Senja Pollak**[2]
[1]Jožef Stefan International Postgraduate School
[2]Jožef Stefan Institute, Ljubljana, Slovenia
[3]Computational Linguistics Lab, Queen Mary University of London, UK
andraz.pelicon@ijs.si, r.shekhar@qmul.ac.uk, matej.martinc@ijs.si
blaz.skrlj@ijs.si, m.purver@qmul.ac.uk, senja.pollak@ijs.si

Abstract

We present a system for zero-shot cross-lingual offensive language and hate speech classification. The system was trained on English datasets and tested on a task of detecting hate speech and offensive social media content in a number of languages without any additional training. Experiments show an impressive ability of both models to generalize from English to other languages. There is however an expected gap in performance between the tested cross-lingual models and the monolingual models. The best performing model (offensive content classifier) is available online as a REST API.

1 Introduction

Recent years have seen a dramatic improvement in natural language processing, with machine learning systems outperforming human performance on a number of benchmark language understanding tasks (Wang et al., 2019). This impressive achievement is somewhat tempered by the fact that a large majority of these systems work only for English, while other less-resourced languages are neglected due to a lack of training resources. On the other hand, another recent development is the introduction of systems capable of zero-shot cross-lingual transfer learning by leveraging multilingual embeddings (Artetxe and Schwenk, 2019). These systems can be trained on a language with available resources and employed on a less-resourced language without any additional language specific training.

In this study we present an offensive language classifier available through a REST API which leverages the cross-lingual capabilities of these systems. Due to the exponential growth of social media content, the amount of offensive language and hate speech has seen a steep increase and its identification and removal is no longer manageable by traditional manual inspection of the content (Schmidt and Wiegand, 2017). As a consequence, there is a need for a general model that could be used in content filtering systems to automatically detect such discourse.

Since the majority of research in the area of offensive language and hate speech detection is currently done in monolingual settings, we performed a preliminary study to assess the feasibility of the proposed zero-shot cross-lingual transfer for this task. Two approaches are tested in this study. The first uses multilingual Bidirectional Encoder Representations from Transformers (BERT, Devlin et al., 2019). The second uses Language-Agnostic SEntence Representations (LASER, Artetxe and Schwenk, 2019), a system built specifically for zero-shot cross-lingual transfer using multilingual sentence embeddings. Our best performing model is available online and can be used for detecting offensive content in less-resourced languages with no available training data.

2 Related work

The large majority of research on hate speech is monolingual, with English still the most popular language due to data availability (Wulczyn et al., 2017; Davidson et al., 2017), and a number of English-only shared tasks organized on the topic of hate or offensive speech (e.g., OffenseEval, Zampieri et al., 2019b). Lately, the focus has been shifting to other languages, with several shared tasks organized that cover other languages besides English, e.g. OffenseEval 2020 (Zampieri et al., 2020), EVALITA 2018 (Bai et al., 2018) and GermEval 2018 (Wiegand et al., 2018).

Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation, pages 30–34
April 19, 2021

For example, the EVALITA 2018 shared task (Bai et al., 2018) covered hate speech in Italian social media, the GermEval 2018 (Wiegand et al., 2018) shared tasks explored automatic identification of offensive German Tweets, and Semeval 2019 task 5 (Basile et al., 2019) covered detection of hate speech against immigrants and women in Spanish and English Twitter. Schmidt and Wiegand (2017); Poletto et al. (2020); Vidgen and Derczynski (2020) provide excellent surveys of recent hate speech related datasets.

Ousidhoum et al. (2019) conduct multilingual hate speech studies by testing a number of traditional bag-of-words and neural models on a multilingual dataset containing English, French and Arabic tweets that were manually labeled with six class hostility labels (abusive, hateful, offensive, disrespectful, fearful, normal). They report that multilingual models outperform monolingual models on some of the tasks. Shekhar et al. (2020) study multilingual comment filtering for newspaper comments in Croatian and Estonian.

Another multilingual approach was proposed by Schneider et al. (2018), who used multilingual MUSE embeddings (Lample et al., 2018) in order to extend the GermEval 2018 German train set with more English data. They report that no improvements in accuracy were achieved with this approach.

Cross-lingual hate speech identification is even less researched than the multilingual task. The so-called *bleaching* approach (van der Goot et al., 2018) was used by Basile and Rubagotti (2018) to conduct cross-lingual experiments between Italian and English at EVALITA 2018 misogyny identification task. The only other study we are aware of is a very recent study by Pamungkas and Patti (2019) proposing an LSTM joint-learning model with multilingual MUSE embeddings. Google Translate is used for translation in order to create a bilingual train and test input data. Bassignana et al. (2018) report that the use of a multilingual lexicon of hate words, HurtLex, slightly improves the performance of misogyny identification systems. Closest to our work is that of Glavaš et al. (2020), who propose a dataset called XHATE-999 to evaluate abusive language detection in a multi-domain and multilingual setting.

3 Dataset Description

As an English (EN) training set for *offensive language* classification, we used the training subset of the OLID dataset (Zampieri et al., 2019a). The trained models were evaluated on the test subset of the OLID dataset using their official gold labels and on the test subset of the GermEval 2018 dataset (Wiegand et al., 2018), which also contains manually labeled tweets. Both datasets use hierarchical annotation schemes for annotating hate speech content. For our purposes, we employed only the annotations on the first level which classify tweets into two classes, offensive and not offensive.

We trained the *hate speech* classifiers on the English training set from the HatEval dataset (Basile et al., 2019). For evaluation, we used the English and Spanish (ES) test sets from the HatEval competition, the German (DE) IGW hate speech dataset (Ross et al., 2016), an Indonesian (ID) hate speech dataset (Ibrohim and Budi, 2019) and the Arabic (AR) hate speech dataset LHSAB (Mulki et al., 2019). Each of the test datasets had binary labels that denoted the presence or absence of hate speech, except for the Arabic test set, which modeled hate speech as a three-class task, with labels denoting absence of hate speech, abusive language and hateful language. Since the authors themselves acknowledge there is a fine line between abusive and hateful language, we felt confident to join them into one class that denotes the presence of hate speech in a tweet. Tweets in the German IGW dataset included hate speech labels from two annotators and no common label, so we decided to evaluate only on those tweets where the two annotators agreed. The statistics of the datasets that were used in this study are reported in Table 1.

4 Classification models and methodology

Our models were trained and evaluated on two distinct albeit similar tasks, namely offensive language classification and hate speech detection, using two different approaches.

In the first approach, we tested the multilingual version of BERT to which we attached a classification layer with a softmax activation function. The model was fine-tuned on the chosen training datasets for 20 epochs. We limited the input sequence to 256 tokens and used a batch size of 32 and a learning rate of 2e-5. No additional hyperparameter tuning was performed.

Our second approach was using the pre-trained

	OLID (EN)	GermEval (DE)	HatEval (EN)	HatEval (ES)	IGW (DE)	ID	L-HSAB (AR)
# documents	14,100	8,541	13,000	6,600	541	13,169	5,846
Majority class	67%	66%	60%	60%	85%	57.77%	62.43%
Minority class	33%	34%	40%	40%	15%	43.23%	37.55%

Table 1: Dataset statistics.

LASER model and training a multilayer perceptron classifier with RELU activation function on top of that. To train the models we used the batch size of 32 and a learning rate of 0.001.

5 Results

The results for both tasks together with the majority baselines and the results reported in the literature are presented in Table 2. In the offensive language classification task, our best model (BERT) achieved an F1 score of 82.63 on the English test set, which is on par with the reported results achieved by monolingual classifiers (Zampieri et al., 2019b). When evaluated on the German dataset, we observe a considerable drop in performance compared to the reported results (Wiegand et al., 2018), however, it still achieves a solid F1 score of 70.67, which indicates its ability to generalize to languages it has not seen during training.

In the hate speech classification task, the two models are comparable, with LASER outperforming BERT on the Arabic and Spanish datasets. Overall, the scores for the hate speech classification task proved to be considerably lower for both models as well as lower than the reported results in the monolingual experiments (Basile et al., 2019; Ibrohim and Budi, 2019). Nevertheless, the results again indicate the ability of both models to generalize from English to other languages, as our models perform better than the majority baseline classifiers in terms of macro-averaged F1 score on all the datasets. It should be noted that the performance between our models and the reported performance on the Indonesian and Arabic datasets are not directly comparable as the original training and testing splits from the literature are not available. Therefore, our models were tested on different test splits.

6 Web API design

The best performing cross-lingual model, multilingual BERT for offensive language classification, was implemented as a REST web service in the Flask framework. The design of the web service allows us to easily update the current model with a new version trained on additional data in the future. The web service can be reached programmatically through the endpoint at `http://classify.ijs.si/ml_hate_speech/ml_bert` or through a demo browser-based interface at the URL `http://classify.ijs.si/embeddia/offensive_language_classifier`. The interface is designed for mobile devices and supports most popular screen sizes. It consists of an input area where users can input their sentence and submit it for classification. The classification results as well as the confidence score of the classifier are then displayed under the input area.

7 Conclusion and future work

In the course of this study, we tested the performance of two multilingual models, BERT and LASER, in zero-shot offensive language and hate speech detection. The results for the offensive language classification task show that even in the multilingual setting the BERT-based classifier achieves results comparable to the monolingual classifiers on English language data and solid performance on the German dataset. On the other hand, hate speech classification still proves to be a hard task for the multilingual classifiers as they achieve considerably lower scores on all languages compared to reported results. Nevertheless, both models show an impressive ability to generalize over languages they have not seen during fine-tuning. We implemented the best performing model, multilingual BERT for offensive language classification, as a REST web service. In the future, we plan to perform similar experiments with other multilingual language models, namely the XLM-R models (Conneau et al., 2019), which show increased performance in standard benchmark tasks compared to multilingual BERT, and the recently released CroSloEngual-BERT (Ulčar and Robnik-Šikonja, 2020).

While all datasets used in this study contain social media posts labeled for hate speech or of-

	Cross-lingual hate speech classification									
	Accuracy					F1-macro				
Model	EN	ES	DE	ID	AR	EN	ES	DE	ID	AR
LASER	**0.5241**	**0.6562**	0.5041	0.5755	**0.7013**	0.4994	0.6538	0.4630	**0.5172**	0.5500
BERT	0.5091	0.6313	**0.6369**	0.5823	0.6264	0.4341	0.5839	**0.6886**	0.4603	0.5033
Reported	/	/	/	**0.7353***	**0.9060***	**0.6510**	**0.7300**	/	/	**0.8930***
Majority	0.6000	0.6000	0.8500	0.5800	0.6200	0.3600	0.3700	0.4600	0.3700	0.3800
	Cross-lingual offensive language classification									
LASER	0.7500	/	0.7129	/	/	0.6823	/	0.6508	/	/
BERT	**0.8279**	/	**0.7148**	/	/	0.8263	/	0.7067	/	/
Reported	/	/	/	/	/	**0.829**	/	**0.7677**	/	/
Majority	0.6700	/	0.6600	/	/	0.4200	/	0.4000	/	/

Table 2: Results of the hate speech classification task (models trained on the English hatEval dataset) and offensive language classification task (models trained on the English OLID dataset) in comparison to the monolingual results as reported in the literature. The forward slash ('/') denotes results which are not reported in the literature. Figures marked with * denote results obtained on a different test split.

fensive language, there are still some differences in the way the data was labeled and collected, as each dataset was collected by a different research team. Therefore, some compromises had to be made in the course of this study to consolidate the datasets as best as possible. In order to better control for such variables, we would like to perform our experiment on the recently released XHate-999 dataset which contains instances in six diverse languages that were collected and annotated by the same research team using a unified annotation process. Given the fact we are working with relatively well-resourced languages, another future endeavour would be to also inspect the differences in cross-lingual model performance between zero-shot and few-shot testing scenarios. Finally, we plan on improving the performance of the model specifically on the task of hate speech classification, and update the existing web service.

8 Acknowledgements

This research is supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). The work of AP was funded also by the European Union's Rights, Equality and Citizenship Programme (2014-2020) project IMSyPP (Innovative Monitoring Systems and Prevention Policies of Online Hate Speech, grant no. 875263). The results of this publication reflect only the authors' views and the Commission is not responsible for any use that may be made of the information it contains. MP was also funded by the UK EPSRC under grant EP/S033564/1. We acknowledge also the funding by the Slovenian Research Agency (ARRS) core research programme Knowledge Technologies (P2-0103).

References

M. Artetxe and H. Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the ACL*, 7:597–610.

X. Bai, F. Merenda, C. Zaghi, T. Caselli, and M. Nissim. 2018. RuG@EVALITA 2018: Hate speech detection in Italian social media. *EVALITA Evaluation of NLP and Speech Tools for Italian*, 12:245.

A. Basile and C. Rubagotti. 2018. CrotoneMilano for AMI at Evalita2018. a performant, cross-lingual misogyny detection system. In *EVALITA@ CLiC-it*.

V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. Rangel Pardo, P. Rosso, and M. Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proc. SemEval*.

E. Bassignana, V. Basile, and V. Patti. 2018. Hurtlex: A multilingual lexicon of words to hurt. In *5th Italian Conference on Computational Linguistics, CLiC-it 2018*.

A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

T. Davidson, D. Warmsley, M. Macy, and I. Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proc. ICWSM*.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL-HLT*.

G. Glavaš, M. Karan, and I. Vulić. 2020. XHate-999: Analyzing and detecting abusive language across domains and languages. In *Proceedings of the 28th International Conference on Computational Linguistics*.

R. van der Goot, N. Ljubešić, I. Matroos, M. Nissim, and B. Plank. 2018. Bleaching text: Abstract features for cross-lingual gender prediction. In *Proc. ACL*.

M. O. Ibrohim and I. Budi. 2019. Multi-label hate speech and abusive language detection in Indonesian Twitter. In *Proc. 3rd Workshop on Abusive Language Online*.

G. Lample, A. Conneau, L. Denoyer, and M. Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *Proc. ICLR*.

H. Mulki, H. Haddad, C. Bechikh Ali, and H. Alshabani. 2019. L-HSAB: A Levantine Twitter dataset for hate speech and abusive language. In *Proc. 3rd Workshop on Abusive Language Online*.

N. Ousidhoum, Z. Lin, H. Zhang, Y. Song, and D.-Y. Yeung. 2019. Multilingual and multi-aspect hate speech analysis. In *Proc. EMNLP-IJCNLP*.

E. W. Pamungkas and V. Patti. 2019. Cross-domain and cross-lingual abusive language detection: A hybrid approach with deep learning and a multilingual lexicon. In *Proc. ACL Student Research Workshop*.

F. Poletto, V. Basile, M. Sanguinetti, C. Bosco, and V. Patti. 2020. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, pages 1–47.

B. Ross, M. Rist, G. Carbonell, B. Cabrera, N. Kurowsky, and M. Wojatzki. 2016. Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis. In *Proc. NLP4CMC III: 3rd Workshop on Natural Language Processing for Computer-Mediated Communication*.

A. Schmidt and M. Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proc. 5th International Workshop on Natural Language Processing for Social Media*.

J. M. Schneider, R. Roller, P. Bourgonje, S. Hegele, and G. Rehm. 2018. Towards the automatic classification of offensive language and related phenomena in German tweets. In *14th Conference on Natural Language Processing KONVENS 2018*.

R. Shekhar, M. Pranjić, S. Pollak, A. Pelicon, and M. Purver. 2020. Automating News Comment Moderation with Limited Resources: Benchmarking in Croatian and Estonian. *Journal for Language Technology and Computational Linguistics (JLCL)*, 34(1).

M. Ulčar and M. Robnik-Šikonja. 2020. FinEst BERT and CroSloEngual BERT. In *International Conference on Text, Speech, and Dialogue*.

B. Vidgen and L. Derczynski. 2020. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *Plos one*, 15(12):e0243300.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proc. ICLR*.

M. Wiegand, M. Siegel, and J. Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language.

E. Wulczyn, N. Thain, and L. Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proc. WWW*.

M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar. 2019a. Predicting the type and target of offensive posts in social media. In *Proc. NAACL-HLT*.

M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar. 2019b. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proc. SemEval*.

M. Zampieri, P. Nakov, S. Rosenthal, P. Atanasova, G. Karadzhov, H. Mubarak, L. Derczynski, Z. Pitenis, and c. Çöltekin. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of SemEval*.

Appendix B: Investigating Cross-Lingual Training for Offensive Language Detection

PeerJ Computer Science

Investigating cross-lingual training for offensive language detection

Andraž Pelicon[1,2], Ravi Shekhar[3], Blaž Škrlj[1,2], Matthew Purver[1,3] and Senja Pollak[1]

[1] Jožef Stefan Institute, Ljubljana, Slovenia
[2] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[3] Queen Mary University of London, London, United Kingdom

ABSTRACT

Platforms that feature user-generated content (social media, online forums, newspaper comment sections etc.) have to detect and filter offensive speech within large, fast-changing datasets. While many automatic methods have been proposed and achieve good accuracies, most of these focus on the English language, and are hard to apply directly to languages in which few labeled datasets exist. Recent work has therefore investigated the use of *cross-lingual transfer learning* to solve this problem, training a model in a well-resourced language and transferring to a less-resourced target language; but performance has so far been significantly less impressive. In this paper, we investigate the reasons for this performance drop, via a systematic comparison of pre-trained models and intermediate training regimes on five different languages. We show that using a better pre-trained language model results in a large gain in overall performance and in zero-shot transfer, and that intermediate training on other languages is effective when little target-language data is available. We then use multiple analyses of classifier confidence and language model vocabulary to shed light on exactly where these gains come from and gain insight into the sources of the most typical mistakes.

Subjects Computational Linguistics, Data Mining and Machine Learning, Natural Language and Speech
Keywords Cross-lingual models, Transfer learning, Intermediate training, Offensive language detection, Deep learning

Submitted 11 November 2020
Accepted 2 May 2021
Published 25 June 2021

Corresponding author
Andraž Pelicon,
Andraz.Pelicon@ijs.si

Academic editor
Robertas Damaševičius

Additional Information and Declarations can be found on page 32

DOI **10.7717/peerj-cs.559**

INTRODUCTION

The massive growth of social media in the last two decades has changed the way we communicate with each other. On the one hand, it allows people worldwide to connect and share knowledge; but on the other, there is a corresponding increase in the negativity to which they can be exposed. Offensive language and hate speech are major concerns on social media, and result in poor psychological well-being, hate crime, and minority group prejudice in both virtual and local communities (*Blair, 2019*; *Gagliardone et al., 2015*). As an extreme example, social media posts were one reason to incite violence against Rohingya Muslims in Myanmar in 2017 (*Beyrer & Kamarulzaman, 2017*; *Stevenson, 2018*; *Subedar, 2018*).

There is therefore a growing need to moderate these platforms to minimize hate speech. Platforms like Facebook, Twitter, and YouTube have started taking the necessary steps to monitor their platforms using manual moderation and automated detection (*Simonite,*

How to cite this article Pelicon A, Shekhar R, Škrlj B, Purver M, Pollak S. 2021. Investigating cross-lingual training for offensive language detection. **PeerJ Comput. Sci. 7:e559 DOI 10.7717/peerj-cs.559**

2020; Lomas, 2015). At the same time, countries like Germany (Lomas, 2017) and the UK (Morgan, 2020) are creating regulations to hold social media platforms accountable. However, with billions of messages posted daily on social media platforms, it is nearly impossible to do this manually, and automatic methods are becoming important. Multiple datasets (e.g., Davidson et al., 2017; Zampieri et al., 2019a; Ljubešić, Fišer & Erjavec, 2019), shared tasks (e.g., Wiegand, Siegel & Ruppenhofer, 2018; Zampieri et al., 2020a) and models (e.g., Salminen et al., 2018; Farha & Magdy, 2020; Gao & Huang, 2017; Zampieri et al., 2020a) have been proposed for several languages. However, so far, good accuracy in automatic detection depends upon the availability of substantial, well-labelled datasets: in many domains and in many languages, this is not the case.

A common theme across recent work in NLP which promises to reduce the requirement for such task-specific labeled data is the use of *transfer learning* (see e.g., Ruder, 2019). Typically, in this approach, a large pre-trained language model (LM) is learned using a general *source* task (e.g., masked language modeling or next sentence prediction) over a very large amount of easily obtained unlabeled data. This pre-trained LM—which contains a lot of information about word meaning and dependencies—can then be fine-tuned on the *target* NLP task (e.g., hate speech detection, question answering etc.), requiring only a smaller labeled dataset to achieve state-of-the-art performance (see e.g., Devlin et al., 2019).

While most of this research is focused on the English language only, the principle extends to transfer between languages, and recent work in *cross-lingual transfer* leverages datasets in multiple languages to provide pre-trained models with multilingual embeddings (Artetxe & Schwenk, 2019; Devlin et al., 2019). For example, Devlin et al. (2019) propose a multilingual version of BERT, called mBERT, trained on 104 languages, in which the representations seem to capture significant syntactic and semantic information across languages (Pires, Schlinger & Garrette, 2019). These pre-trained LMs can therefore be trained on a language with available resources and employed on a less-resourced target language without additional language-specific training. This can help alleviate the data availability gap between high-resourced and less-resourced languages: for example, Leite et al. (2020) perform zero-shot transfer from English to Brazilian Portuguese for toxic comment detection. Most such studies are however restricted to evaluating zero-shot transfer from one language to one other only, and using only one multilingual pre-trained LM. Furthermore, several studies (Stappen, Brunn & Schuller, 2020; Leite et al., 2020), including our own initial work (Pelicon et al., 2021), suggest that this *zero-transfer* approach to multilingual training does not achieve performance comparable to systems trained on the actual target language data. As such, some amount of data in the target language is still preferred and may be needed for good accuracy. However, it is not clearly understood how exactly the amount of data affects this requirement and the performance of the final models.

Another question that remains largely unexplored is whether this data shortage problem can instead be addressed by using training data in one or several other non-target languages. An *intermediate training* mechanism has been proposed (Yogatama et al., 2019; Wang et al., 2019a; Pruksachatkun et al., 2020; Vu et al., 2020) to reduce the need for large

scale data for all tasks in all languages. In the intermediate training step, instead of fine-tuning the LM directly on the target language task, it is first trained on a similar task using the same or different language data. *Pruksachatkun et al. (2020)* show that performing intermediate training using English data improves the multiple XTREME benchmark tasks (*Hu et al., 2020*). *Robnik-Sikonja, Reba & Mozetic (2020)* perform sentiment classification using training data from both target language and several non-target languages. However, this work is evaluated only in a setting where all available target language data is used for training: it is therefore hard to tell whether and how the benefit of intermediate training depends on how much target data is available. *Stappen, Brunn & Schuller (2020)* investigate this, via an analysis of cross-lingual capabilities of their hate speech model in which they first train a model in one language and then progressively add data in the target language. However, their analysis is performed only on one pair of languages. From these studies alone it is therefore not yet clear how much of the performance gap is due to the pre-trained model and its properties, and how much to the training regime, choice of intermediate languages and relative amount of data available.

In this work we perform a thorough analysis of the feasibility of training models that leverage multilingual representations with non-target language data. Specifically, we address the following research questions:

- *Effect of pre-trained LM:* How does the choice of multilingual pre-trained language model affect performance?
- *Effect of intermediate training:* Where little or no target language training data is available, when and by how much does intermediate training in a different language boost performance?
- *Data hunger of the model:* How does performance depend on the amount of intermediate and/or target language data?

We used five hate speech datasets in different languages, namely Arabic, Croatian, German, English, and Slovenian. All these languages are included in the standard pre-trained mBERT model. Arabic, German and English were chosen for their range of similarity: while German is fairly similar to English, sharing many syntactic and vocabulary features, Arabic is dissimilar to both, with very different linguistic features, an entirely different alphabet, and written right-to-left rather than left-to-right. Croatian and Slovenian were then chosen for being less-resourced, for representing a mid-point in similarity (being Slavic languages, they are less similar to English than German is, but more so than Arabic), and because they are included in a more specific trilingual Croatian-Slovenian-English pre-trained language model based on BERT architecture (*Ulčar & Robnik-Šikonja, 2020*, see "Background and Related Work"). This selection allows us to test a range of hypotheses, including that intermediate training may be more useful for more similar languages and that more specific LMs transfer better. We show that cross-lingual transfer can be useful for the offensive language detection task, and that using a more specific LM significantly improves performance for Croatian and Slovenian, even in the low data regime. We perform multiple analyses to shed light on the behavior of

the models, and use visualization techniques to try and interpret the inner workings of our fine-tuned models.

The paper is organized as follows; first, in "Background and Related Work", we start by providing a summary of offensive language detection, the use of different pre-trained language models, and intermediate training. In "Method and Datasets", we describe our experimental pipeline, the dataset used, and model architecture. "Quantitative Results" presents our experiments and quantitatively answers our research questions. "Analysis and Qualitative Results" provides insight into the results using different analyses and some qualitative results. "Conclusion" concludes our contribution. The paper also contains an "Appendix" with additional detailed experimental results. The code and data splits for the experiments are made available on GitHub (https://github.com/EMBEDDIA/cross-lingual_training_for_offensive_language_detection).

BACKGROUND AND RELATED WORK

In this section we present an overview of the state of the art in offensive language detection, first reviewing defining the task and reviewing available datasets (Offensive Language Detection: Task and Datasets), and next describing current approaches to automatic detection, explaining their use of pre-trained language models (Automatic Detection and Pre-Trained Models). We then discuss approaches to multilinguality and cross-lingual training (Multilingual and Cross-lingual Approaches), and explain in detail the technique of intermediate training that we investigate here (Intermediate Training).

Offensive language detection: task and datasets

Automatically detecting hate or offensive language is an increasingly popular task, with many public datasets, shared tasks, and models proposed to tackle it (see *Schmidt & Wiegand, 2017*; *Poletto et al., 2020*; *Vidgen et al., 2020*; *Vidgen & Derczynski, 2020*, for recent surveys). The exact definition of the categories annotated in these tasks varies, but they generally include threats, abuse, hate speech and offensive content. These terms are often used interchangeably, with some (particularly *hate speech*) often used to cover multiple categories. Exact definitions of the individual categories also vary with task and dataset. In this work, we use *offensive speech* as a generic term. The task is usually defined as a classification task, i.e., for a given text, determine if it is hate speech or not. Some tasks also try to classify at finer-grained levels and treat the task as a multi-class problem.

Datasets and languages

Most research on offensive language detection is monolingual, and English is still the most popular language, at least partly due to data availability (*Wulczyn, Thain & Dixon, 2017*; *Golbeck et al., 2017*; *Davidson et al., 2017*; *Vidgen et al., 2020*). Most data is collected from social media platforms (such as Twitter (*Davidson et al., 2017*), Facebook (*Ljubešić, Fišer & Erjavec, 2019*)), newspaper comments (*Gao & Huang, 2017*), YouTube (*Obadimu et al., 2019*), and Reddit (*Qian et al., 2019*). Lately, however, the focus has been shifting to other languages, with several shared tasks organized that cover other languages

besides English, including EVALITA 2018 (*Bai et al., 2018*), GermEval 2018 (*Wiegand, Siegel & Ruppenhofer, 2018*) and SemEval 2019 Task 5 on Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter (*Basile et al., 2019*). The OffensEval 2020 shared task (*Zampieri et al., 2020a*) also featured five languages: Arabic, Danish, English, Greek, Turkish. Some other non-English datasets for offensive language exist: *Ibrohim & Budi (2018)* annotated Indonesian tweets for abusive language, and *Mubarak, Darwish & Magdy (2017)* annotated abusive Arabic tweets. For Spanish, *Plaza-Del-Arco et al. (2020)* provide tweet collection annotated for misogyny and xenophobia, while *Leite et al. (2020)* provide toxic tweet collection in Brazilian Portuguese. *Mathur et al. (2018)* and *Chopra et al. (2020)* present data in Hinglish (spoken Hindi mixed with English written using the Roman script). The HASOC dataset (*Mandl et al., 2019*) is in English, German and Hindi, with both tweets and Facebook comments. *Ljubešić, Erjavec & Fišer (2018)* and *Shekhar et al. (2020)* provide data from Croatian newspaper comment sections.[1]

[1] A comprehensive list of relevant datasets is available online at http://hatespeechdata.com/.

Automatic detection and pre-trained models

A range of machine learning methods have been proposed to address the task, including logistic regression (*Davidson et al., 2017*; *Pedersen, 2020*), Naive Bayes (*Shekhar et al., 2020*), support vector machines (*Salminen et al., 2018*), and deep learning (DL) (*Zampieri et al., 2020a*). Most approach the problem as one of text classification, but some try to improve results via the addition of other data: *Gao & Huang (2017)* use the username and the title of the article as context to perform the task, while *Farha & Magdy (2020)* use a multi-task approach, and *Salminen et al. (2020)* develop a taxonomy of hate speech types with corresponding multiple models. Most recent approaches are DL-based, and a general trend in this direction is the use of pre-trained language models (LMs).
The availability of large amounts of data, computational resources and the recently introduced Transformer architecture (*Vaswani et al., 2017*) have resulted in a large number of such pre-trained LMs, e.g., BERT (*Devlin et al., 2019*), RoBERTa (*Liu et al., 2019*) and others. These models are generally used by taking the pre-trained LM model weights as initialization, adding a task-specific classifier layer on top, and fine-tuning it using task-specific data. Variants of this approach have been shown to achieve the state of the art performance on multiple tasks like question-answering (*Rajpurkar et al., 2016*), the GLUE (*Wang et al., 2018*) and SuperGLUE (*Wang et al., 2019b*) benchmarks, as well as hate speech detection (see e.g., *Liu, Li & Zou, 2019*). In the OffensEval-2020 shared task (*Zampieri et al., 2020a*), most of the best-performing models use a variant of this approach.

Multilingual and cross-lingual approaches

All these approaches, however, rely on suitable labeled training datasets in the target language. As explained in "Offensive Language Detection: Task and Datasets", language coverage is increasing, but no datasets currently give (or can hope to give) resources

for all languages, and any work in less-resourced languages will therefore require the development of new datasets from scratch. There is therefore significant interest in *cross-lingual* approaches to hate speech identification, in which a model for a chosen *target* language is trained using data in one or more different, better-resourced *source* languages.

Basile & Rubagotti (2018) conduct cross-lingual experiments between Italian and English on the EVALITA 2018 misogyny identification task, using the so-called *bleaching* approach (*van der Goot et al., 2018*), which aims to transform lexical strings into a set of abstract features in order to represent textual data in a language-agnostic way. While this approach shows a drop in performance in a monolingual setting, it outperforms the standard lexical approaches in a cross-lingual setting. More recent work uses neural networks: *Pamungkas & Patti (2019)* use a LSTM joint-learning model with multilingual MUSE embeddings, which are trained from parallel corpora in order to give cross-lingual representations (*Lample et al., 2018*). This showed improvement in a cross-lingual setting over a SVM with unigram features. However, cross-lingual models generally seem to perform worse than monolingual ones. *Leite et al. (2020)* tested monolingual and cross-lingual models based on multilingual BERT on Spanish and Portuguese data; the monolingual models outperformed their cross-lingual counterparts. *Schneider et al. (2018)* used multilingual MUSE embeddings to extend the GermEval 2018 German training set with more English data, but saw no improvement in performance. *Stappen, Brunn & Schuller (2020)* extended the original XLM architecture to a cross-lingual setting, and evaluated it in zero-shot (i.e., without any data in the target language) and few-shot (small amounts of target language data) settings, and found that even a small amount of target language data substantially improves model performance over the zero-shot setting.

Several questions remain unanswered, though. First, it is not yet clear how general this performance drop is across languages; *Stappen, Brunn & Schuller (2020)*, for example, look at only one language pair, namely English and Spanish. In this paper, we therefore examine a broader range of languages. Another is the effect of the pre-trained LM used. Most current cross-lingual approaches are based on multilingual versions of the pre-trained LMs introduced above, such as multilingual BERT (mBERT, *Devlin et al., 2019*) and XLM-R (*Conneau et al., 2020*); as these are pre-trained on large multilingual corpora, their representations can transfer well between the languages seen in pre-training, and cross-lingual effects within these can be achieved by fine-tuning on a source language dataset and testing on a different target language. However, while these LMs perform reasonably well across a range of languages and tasks, they perform less well on a given domain or language than a model pre-trained for that specific domain (e.g., *Lee et al., 2020*, for biomedicine) or language (e.g., *Martin et al., 2020*, for French). *Ulčar & Robnik-Šikonja (2020)* provide two tri-lingual BERT models, FinEstBERT (Finnish/Estonian/English) and CroSloEngualBERT (Croatian/Slovenian/English), and show that they perform better in those languages than the more general mBERT on several tasks like NER, POS-tagging and dependency parsing. We might therefore expect LMs with more specific

language combinations to perform better at cross-lingual transfer within those combinations, and this is another question we investigate here.

Intermediate training

Another question is the effect of the choice and combination of source vs target language data when fine-tuning the pre-trained LM. The general mechanism in use here is often called *intermediate training*: starting with a pre-trained LM, first training on a similar source (or rather, in this setting, *intermediate*) task, and only then training on the desired target task. Most work in this direction examines the effect of intermediate training on a source task different from the target task (*Yogatama et al., 2019*; *Wang et al., 2019a*; *Pruksachatkun et al., 2020*; *Vu et al., 2020*). *Yogatama et al. (2019)* explore the transferability of linguistic knowledge in the LM to the target task: while some knowledge is transferred, fine-tuning is still needed to perform the target task, and the fine-tuned model is less transferable to the same task on different datasets. *Wang et al. (2019a)* conducted 17 instances of intermediate training on ELMo and BERT models on the GLUE benchmark tasks, finding that intermediate training doesn't always help with target tasks. Surprisingly, they found no clear correlation between the intermediate task data size and fine-tuned target task performance. *Pruksachatkun et al. (2020)* also performed an extensive study of intermediate training using RoBERTa (*Liu et al., 2019*); consistent with *Wang et al. (2019a)*, they also found no impact of intermediate task dataset size. In general, having high-level inference (e.g., co-reference resolution) and commonsense reasoning (e.g., QA) tasks as the intermediate task is helpful. In contrast to other work, *Vu et al. (2020)* show that intermediate training has a more significant effect on performance, and tested different settings to understand the impact of intermediate and target dataset size. The performance gain is highest when there is limited target training data; and the transferability of knowledge from intermediate to the target task is more dependent on the similarity between the intermediate and target tasks and datasets. *Pelicon et al. (2020)* used a sentiment classification task as intermediate task to boost the performance of the target task of news sentiment classification, with consistent findings. *Lin et al. (2019)* proposed a systematic way to transfer knowledge from one language to another, via a mechanism to select the best language pair for the transfer of knowledge.

In the domain of offensive language detection, *Stappen, Brunn & Schuller (2020)*'s cross-lingual experiments (see "Multilingual and Cross-lingual Approaches" above) can also be seen as an example of intermediate training, first fine-tuning with data in a language that was different from the target language, and then with differing amounts of data in the target language. They found that performance improves only in the case of small amounts of target data. As noted above, though, they investigated only one language pair (English/Spanish), and used only a general mBERT LM. Here, we attempt a more systematic and wider investigation of different intermediate training regimes, with different language pairs, and different pre-trained LMs.

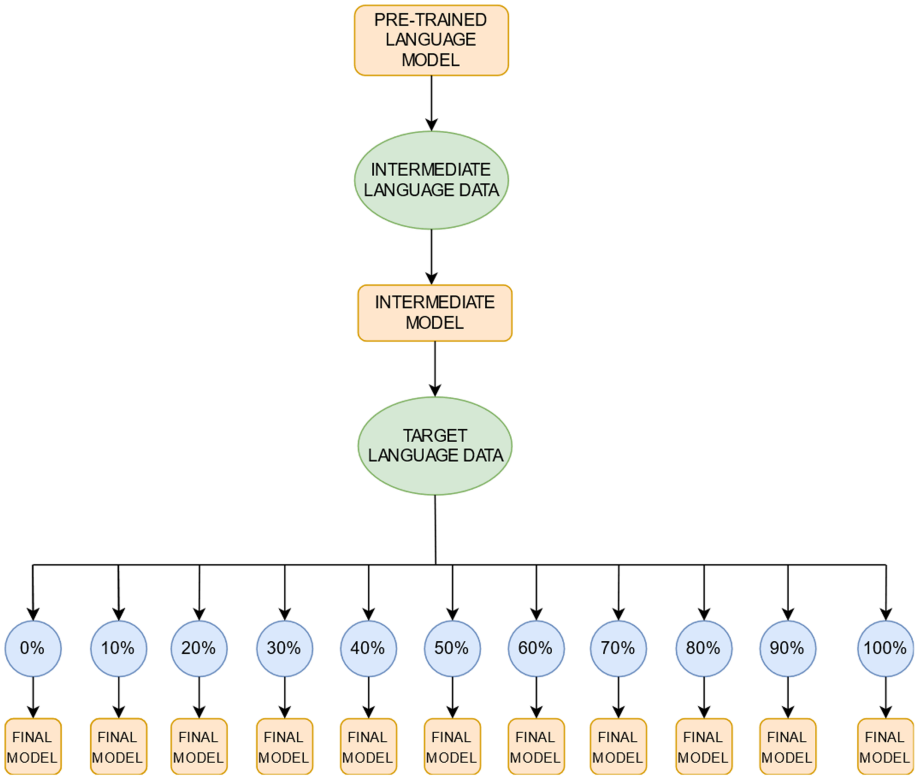


Figure 1 A schematic illustration of the training regime. We first select a *pre-trained* language model; further train it on data in one or more *intermediate* non-target languages to produce an *intermediate* model; then fine-tune the result by progressively adding data in the target language to produce the *final* model with which to evaluate performance. We progressively add data in the target language in 10% increments; the blue circles represent the proportion of target language data we use for training the final models. The step size of 10% was chosen arbitrarily. Note that the 0% setting presents the *zero-shot learning* setting where no target language data is used for fine-tuning and the intermediate model is evaluated directly on the target language data. Full-size DOI: 10.7717/peerj-cs.559/fig-1

METHOD AND DATASETS

In this study, we investigate the effectiveness of cross-lingual training for the problem of hate speech detection. This problem can be modeled as a classification task, formally stated as follows.

Let

$$\mathrm{NN} : \mathbf{X}_l \rightarrow C$$

represent a classifier able to map from the space of text representations (e.g., byte pair encoded inputs) $\mathbf{X}_l$ in a given language l to the set of possible classes C. The purpose of this work is to explore the predictive performance of NN in a cross-lingual setting. Formally, we explore the performance of NN when trained on the space $\mathbf{X}_a$ and tested on $\mathbf{X}_b$, where a and b represent two different languages.

In this section, we describe our experimental setup, datasets, the details of our classification model architecture and optimization, and the evaluation metrics used.

Experimental pipeline

Our experimental pipeline (Fig. 1) consists of three steps: selection of a pretrained language model (LM), intermediate-task training on data in one or more non-target languages, and fine-tuning on a single target-language task. In the last fine-tuning step, we test the effect of variable amounts of target-language training data.

Language model

In order to investigate the effect of the pre-trained LM properties, we use two multi-lingual transformer based models: mBERT, a general model with 104 languages (*Devlin et al., 2019*), and CroSloEngual BERT, hereafter cseBERT, a much more specific model with only three languages (*Ulčar & Robnik-Šikonja, 2020*). All the languages used in the experiments are present in mBERT; three languages (Croatian, Slovenian and English) are present in cseBERT, allowing us to compare its effect on those and on others not included in its pre-training.

Intermediate training

In this step, we perform intermediate-task training of the model on a classification task in one or more non-target languages. We focus on three different languages for intermediate training, namely English, Slovenian and Arabic. English and Slovenian are used because they are used in both mBERT and cseBERT; use Latin script, common for all languages except Arabic; and give two points for comparison of language similarity (Slovenian is more similar to Croatian and less similar to German; English is more similar to German and less to Croatian, as discussed in "Introduction"). Finally, we include Arabic as it is the most dissimilar from all other languages used here, in terms of both linguistic and orthographic features, and is present in mBERT but not in cseBERT. We also test the use of intermediate training on all the languages except for the target language, and call this the *leave-one-(language-)out* (LOO) setting.

Target task fine-tuning

In the final step, we fine-tune our model on the target language task dataset following the standard procedure (*Devlin et al., 2019*). Depending on the configuration of the first two steps, the target task performance can then be observed with the different LMs, and with and without the different intermediate training variants.

Data hunger of the model

To observe how data availability influences the performance on the target language task, we gradually increase the amount of training data for the fine-tuning, from 0% target data (the zero-transfer setting) to 100% target data (the ideal fully-resourced scenario) in steps of 10%. We use this increasing data regime to investigate the following questions. First, does having a better pre-trained LM reduce the amount of target data needed to achieve good performance? Second, to what extent can intermediate training on another language compensate for unavailability of target language data (which would be especially

Table 1 Original dataset sizes and label distribution.

Language	Source	Original size	Not-offensive proportion (%)	Offensive proportion (%)
Croatian (*Shekhar et al., 2020*)	News comment	99,246	50	50
Slovenian (*Ljubešić, Fišer & Erjavec, 2019*)	Facebook	12,400	46	54
English (*Zampieri et al., 2019a*)	Twitter	13,240	67	33
German (*Wiegand, Siegel & Ruppenhofer, 2018*)	Twitter	8,884	67	33
Arabic (*Zampieri et al., 2020a*)	Twitter	7,839	80	20

valuable for less-resourced languages)? Last but not least, we test whether training in intermediate language(s) can boost the performance compared to training only in the target language.

Datasets

We used hate speech and offensive language datasets in five different languages—English, Arabic, Croatian, Slovenian and German (see Table 1)—for intermediate training and fine-tuning:[2]

- **Croatian: 24sata** (*Shekhar et al., 2020*, *Pollak et al., 2021*). This dataset contains reader comments from the Croatian online news media platform 24sata (https://www.24sata.hr/). Each comment is labeled according to 8 rules covering Disallowed content (Spam), Threats, Hate speech, Obscenity, Deception & trolling, Vulgarity, Language, Abuse (see *Shekhar et al., 2020*, for annotation schema details). In this study we used only the Hate speech label, taking all comments without that label as non-hate speech.
- **English: OffensEval 2019** (*Zampieri et al., 2019a*). This dataset contains Twitter posts that are labeled according to a three-level annotation scheme. On the first level, each tweet is labeled as either offensive or not offensive. Those labeled as offensive are then annotated on a second level as either targeted (i.e., directed at a particular individual or group) or untargeted (i.e., containing general profanity). Those labeled as targeted are further labeled on a third level as directed towards a specific individual, group or other entity. For our task we use only the first level (offensive/non-offensive).
- **Slovenian: FRENK** (*Ljubešić, Fišer & Erjavec, 2019*). This dataset contains Facebook posts, and uses a 3-label annotation schema, where each post is annotated as Acceptable, Other offensive (i.e., containing general profanity), Background offensive (i.e., containing insults or profanity targeted at a specific group). The dataset is divided in two parts, one on the topic of migrants and migrations and the other on the topic of LGBT communities. Both parts were collected by the same group following the same procedure. We used both migrant and LGBT datasets together and combine all offensive classes into one class.
- **German: GermEval 2018** (*Wiegand, Siegel & Ruppenhofer, 2018*). This dataset contains Twitter posts labeled on two levels. On the first level, each tweet is labeled as either

[2] All the datasets used in this study were gathered in the course of other studies. For Slovenian the data is not public, but is available upon request from the original authors; for all other languages the datasets are publicly available (see cited references for details), and our GitHub repository (https://github.com/EMBEDDIA/cross-lingual_training_for_offensive_language_detection) provides exact data splits used in our study.

Offensive or Other. Those labeled as Offensive are then labeled on the second level as either Profanity, Abuse or Insult. For our classification task, we use only the first level (offensive/non-offensive).

- **Arabic: OffensEval 2020** (*Zampieri et al., 2020a*). This dataset contains Twitter posts, gathered and annotated by the same team as the OffensEval 2019 English Dataset (see above); it uses the same annotation schema and we treat it in the same way.

Although all the datasets were annotated for hate speech or offensive language detection tasks, the authors employed different annotation schemes due to their domain and specific purposes and phenomena. This reflects the current situation, in which a large number of labeled hate speech datasets are freely available for different languages, but do not share a common annotation procedure. These discrepancies, albeit small, can potentially impact a model's ability to properly converge if one were trying to boost performance using data across several datasets and languages. In this way, our experimental setting reflects this real-world scenario and provides a realistic estimation of the models' behavior.

To deal with the differences in annotations, we consolidated the annotation schemas of different datasets so as to model the problem as a similar binary classification task in each case. For this purpose, we use the first-level annotations of the English, German and Arabic datasets, which label the documents as either offensive or not offensive. For the Slovenian dataset, in which offensive posts are labeled in several categories on one level, we combine the different offensive categories into one offensive class. For the Croatian dataset only the hate speech label is used, as the other categories represent different reasons for blocking comments which may not necessarily include offensive language of any kind.

To minimize the effect of dataset size on the performance of the model, we use the same amount of training data for each language. We reduced the size of all datasets to the size of the smallest dataset in the set, namely the Arabic dataset with 7839 instances, while keeping the class balance the same. We split the resulting datasets into training, validation and test sets in the proportion 80-10-10.[3]

[3] The splits for the English, German, Croatian and Arabic datasets are available on the GitHub repository (https://github.com/EMBEDDIA/cross-lingual_training_for_offensive_language_detection). The code for Slovenian data splits is provided on the same GitHub, however the data itself should be obtained from *Ljubešić, Fišer & Erjavec (2019)*.

Models and optimization

We perform the whole three-step experiment described in "Experimental Pipeline" using a BERT-based language model (mBERT or cseBERT). The representation of the (CLS) token from the last layer of the BERT language model is used as a sentence representation, and passed to a further linear layer with a softmax activation function to perform the classification. The whole model is jointly trained on the downstream task of hate speech detection. Fine-tuning is performed end-to-end. All models were trained for maximum 4 epochs with batch size 16. The best model is selected based on the validation score. We used the Adam optimizer with the learning rate of 2×10^{-5} and learning rate warmup over the first 10% of the training instances. For regularization purposes we used weight decay rate set to 0.01. The same optimization process was used for both the intermediate training and the fine-tuning steps of our training setup. We perform the training of the

models using the HuggingFace Transformers library (*Wolf et al., 2020*). To perform matrix operations in an efficient manner we ensured all inputs were of the same length, first tokenizing all inputs and then setting their maximum length to 256 tokens. Sequences larger than this maximum were shortened, while longer sequences were zero-padded. As is standard with the BERT architecture, each of these models was pre-trained with minimal text preprocessing and comes with its own tokenizer which tokenizes text at word and sub-word levels. We applied the same procedure in the intermediate learning and fine-tuning phases, tokenizing the text input using the default tokenizers that were trained with the mBERT and cseBERT models, with no additional text pre-processing.

Evaluation metrics

Due to imbalance in the dataset, we follow the standard evaluation metrics used in OffensEval (*Zampieri et al., 2019a*) and report the macro-averaged F1 score. To counteract the effect of random initialization of the model, we trained models with three different random seeds and report mean and standard deviations of F1 scores. To qualify the performance with increasing data, we report the area-under-curve (AUC) with respect to the F1-score and data size. For more detailed evaluation information, we also provide two other standard evaluation metrics, macro-averaged recall and precision, again reported as mean and standard deviation over the three training runs with different random seeds. For readability purposes, we present these results in the "Appendix".

To test for statistical significance of differences between results, we use the Mann–Whitney *U* test with a significance level of 0.05. We choose this non-parametric test as it makes no assumptions about normality of distribution and is suitable to be used with a small number of samples (3 runs of each experiment in our case).

QUANTITATIVE RESULTS

In this section, we present quantitative results, and in particular answer the research questions presented in "Introduction" concerning the effects of pre-trained model selection, intermediate training (using one or more additional languages), and amount of target language training data.

Monolingual results

To provide points of comparison, we first give results for the standard monolingual case in which all target-language data is assumed to be available and used in fine-tuning, with no intermediate training; together with baseline results based on the majority class and on random model weight initialization. For the majority class baseline, we simply give all test set examples the same label as the majority class in the training set data. For the random initialization baseline, we attach the pre-trained LM to the randomly initialized classifier layer.

Table 2 shows these results for both mBERT and cseBERT. Random initialization of the model is in most cases similar to the majority class baseline and has very high standard deviation; it allows us to explicitly examine the effect of fine-tuning. As expected, after fine-tuning the model on the entire target-language dataset, the performance of the

Table 2 Comparison of mBERT and cseBERT, fine-tuning on all training data in the target language only (no intermediate training), together with the majority class and randomly initialized models baselines. Values are shown as macro-averaged F1-score with standard deviation. Bold indicates the best performance for each language; † indicates that the difference is statistically significant based on the Mann–Whitney *U* test. For comparison also the following state-of-the-art (SOTA) results are provided: *Shekhar et al. (2020)*[1], *Miok et al. (2021)*[2], *Zampieri et al. (2019b)*[3], *Struß et al. (2019)*[4], *Zampieri et al. (2020b)*[5]. Note however that the SOTA results are based on different data splits. For macro-averaged precision and recall scores, see Tables 10 and 11.

Language	Majority class	mBERT		cseBERT		SOTA
		Random init.	Fine-tuned	Random init.	Fine-tuned	
Croatian	43.72	$49.99_{3.30}$	$71.10_{1.42}$	$45.85_{4.83}$	†$\mathbf{74.98}_{1.06}$	71.78[1]
Slovenian	34.83	$44.33_{6.44}$	$72.73_{0.36}$	$44.94_{3.27}$	†$\mathbf{76.11}_{0.58}$	68.60[2]
English	41.89	$47.72_{3.57}$	$76.63_{1.15}$	$42.32_{9.09}$	$\mathbf{77.10}_{1.34}$	82.90[3]
German	39.46	$31.19_{4.89}$	†$\mathbf{75.90}_{0.38}$	$40.96_{10.60}$	$73.98_{0.98}$	76.95[4]
Arabic	44.32	$50.13_{1.91}$	†$\mathbf{84.62}_{0.19}$	$45.73_{9.26}$	$76.01_{0.61}$	90.17[5]

model is always substantially higher than the majority class and random initialization baselines (for both mBERT and cseBERT). The highest gain over the majority class baseline is observed for Arabic with mBERT, and for Slovenian with cseBERT. The best performances for each language (see bold columns in Table 2) are overall of a similar level to those reported in other work, giving us confidence that we are experimenting with models which approach the monolingual state of the art. Please note, however, that due to resizing of the datasets (as explained in "Datasets") our results were obtained on different train-validation-test splits than the results from related work and are therefore not directly comparable.

Effect of pre-trained LM

Comparing the performance of mBERT and cseBERT (Fine-tuned columns in Table 2), we observe that using cseBERT always outperforms mBERT for the languages cseBERT is pre-trained on (ΔF1 +3.88 Croatian, +3.38 Slovenian, +0.47 English); but performance decreases for languages not used in cseBERT pre-training (ΔF1 −1.92 German, −8.61 Arabic). For English, mBERT and cseBERT performances are very similar.
The improvement in performance in Slovenian and Croatian using cseBERT, which was pre-trained with higher quality resources for Slovenian and Croatian, is consistent with the findings of the authors of cseBERT (*Ulčar & Robnik-Šikonja, 2020*) on a range of tasks. This also suggests that improving the pre-trained models especially benefits less-resourced languages like Slovenian and Croatian. The decrease in performance for Arabic is higher than that for German. This could be attributed to the fact that cseBERT is pre-trained only on languages in Latin script, perhaps resulting in little overlap in sub-word token vocabulary with Arabic. For German, some sub-words will be shared between the languages in the pre-training and testing phases (see "Analysis of Vocabulary Coverage"). However, as the performance of cseBERT is still decent on languages not used in pre-training, the fine-tuning step seems of high importance and the pre-training phase plays only a limited role in these cases.

Table 3 Comparison of intermediate training in a range of non-target languages in zero-shot transfer on the target language data, for mBERT (top) and cseBERT (bottom). TGT: random initialization (no intermediate training, no target fine-tuning). ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then zero-shot transfer on the target language. LOO → TGT: Intermediate training on all non-target languages, then zero-shot transfer on the target language. Values are shown as macro-averaged F1-score with standard deviation. Bold indicates the best performance for each target language and arrows indicate increase/decrease compared to the randomly initialized baseline. For macro-averaged precision and recall scores, see Tables 12 and 13.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	mBERT				
Croatian	$49.99_{1.54}$	$\uparrow 60.30_{1.02}$	$\uparrow 59.97_{0.22}$	$\downarrow 47.98_{0.46}$	$\uparrow \mathbf{62.83}_{0.58}$
Slovenian	$44.33_{1.45}$	$\uparrow \mathbf{59.57}_{0.77}$	–	$\downarrow 35.55_{0.88}$	$\uparrow 47.00_{0.93}$
English	$47.72_{0.90}$	–	$\downarrow 43.28_{1.40}$	$\downarrow 44.11_{0.21}$	$\uparrow \mathbf{49.07}_{0.52}$
German	$\mathbf{31.19}_{1.82}$	$\downarrow 28.43_{1.95}$	$\downarrow \textit{28.01}_{4.41}$	$\downarrow 27.43_{6.63}$	$\downarrow 27.72_{9.72}$
Arabic	$50.13_{2.90}$	$\downarrow 46.00_{2.53}$	$\uparrow \mathbf{59.68}_{2.43}$	–	$\uparrow 56.71_{1.31}$
	cseBERT				
Croatian	$45.85_{9.87}$	$\uparrow \mathbf{67.70}_{0.34}$	$\uparrow 67.56_{0.69}$	$\downarrow 44.51_{0.97}$	$\uparrow 67.12_{0.91}$
Slovenian	$44.94_{1.47}$	$\uparrow \mathbf{63.98}_{0.12}$	–	$\downarrow 34.34_{0.28}$	$\uparrow 58.75_{0.40}$
English	$42.32_{14.15}$	–	$\uparrow 53.61_{0.34}$	$\uparrow 44.67_{1.42}$	$\uparrow \mathbf{60.42}_{0.88}$
German	$\mathbf{40.96}_{5.52}$	$\downarrow 25.69_{1.56}$	$\downarrow 26.20_{0.00}$	$\downarrow 25.83_{0.77}$	$\downarrow 26.63_{0.00}$
Arabic	$\mathbf{45.73}_{6.40}$	$\downarrow 44.97_{3.30}$	$\downarrow 44.97_{4.54}$	–	$\downarrow 44.97_{3.15}$

Effect of intermediate training

As a next research question, we asked whether intermediate training on different languages can boost the classifier performance on the target language. First, we evaluate the effect of intermediate training without fine-tuning on the target language training data: the *zero-shot transfer* scenario. As Table 3 shows, for most cases, intermediate training gives substantial increases over the baseline, except for German and Arabic with cseBERT. This shows that the model learns some useful knowledge from intermediate training and transfers it to the target language task: performances are reasonable in many cases, although they do not reach the levels of the monolingual results of Table 2, confirming the findings of *Stappen, Brunn & Schuller (2020)* and *Leite et al. (2020)*. Again, we see that cseBERT gives better results for its languages (e.g., transfer from English to Croatian and Slovenian) than mBERT, while mBERT does better when Arabic is the target. Encouraged by this result, we test the effect of intermediate training in the well-resourced scenario: fine-tuning the intermediate trained model using all target language task data. Table 4 shows the results of fine-tuning only on target language data (repeated from Table 2), compared to the use of intermediate training using English, Slovenian and Arabic respectively, before fine-tuning in the target language as before. In the last column (LOO+TGT), we include all languages except the target language (LOO) in the intermediate training step.

In most cases, adding one or more languages improves the results (the exceptions being the English target language for mBERT and German target language for cseBERT). However, the gain in performance is not large. In the case of mBERT, the largest gain is

Table 4 Comparison of intermediate training in a range of non-target languages, followed by fine-tuning on all target language data, for mBERT (top) and cseBERT (bottom). TGT: Only fine-tuned on target language (no intermediate training). ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO → TGT: Intermediate training on all non-target languages, then fine-tuning on target language. Values are shown as macro-averaged F1-score with standard deviation. Bold indicates the best performance for each target language and arrows indicate increase/decrease compared to the randomly initialized baseline. For macro-averaged precision and recall scores, see Tables 14 and 15.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	mBERT				
Croatian	$71.10_{1.42}$	$\uparrow 71.96_{1.55}$	$\uparrow \mathbf{72.12}_{0.48}$	$\uparrow 71.88_{0.80}$	$\uparrow 71.43_{0.30}$
Slovenian	$72.73_{0.36}$	$\downarrow 72.33_{1.07}$	–	$\uparrow 73.89_{0.68}$	$\uparrow \mathbf{74.99}_{1.07}$
English	$\mathbf{76.63}_{1.15}$	–	$\downarrow 74.05_{1.01}$	$\downarrow 74.73_{0.31}$	$\downarrow 76.09_{1.04}$
German	$75.90_{0.38}$	$\uparrow \mathbf{76.07}_{0.15}$	$\downarrow 74.46_{0.04}$	$\downarrow 74.90_{1.16}$	$\downarrow 75.02_{0.52}$
Arabic	$84.62_{0.19}$	$\downarrow 84.07_{0.45}$	$\uparrow \mathbf{85.75}_{1.03}$	–	$\uparrow 85.56_{0.53}$
	cseBERT				
Croatian	$74.98_{1.06}$	$\uparrow \mathbf{76.54}_{0.98}$	$\downarrow 74.93_{0.42}$	$\uparrow 75.37_{0.70}$	$\uparrow 76.00_{0.59}$
Slovenian	$76.11_{0.58}$	$\uparrow \mathbf{76.78}_{0.34}$	–	$\downarrow 76.03_{0.44}$	$\uparrow 76.42_{0.31}$
English	$77.10_{1.34}$	–	$\uparrow 77.12_{0.82}$	$\downarrow 77.06_{1.00}$	$\uparrow \mathbf{77.73}_{0.35}$
German	$\mathbf{73.98}_{0.98}$	$\downarrow 71.60_{1.09}$	$\downarrow 69.30_{0.40}$	$\downarrow 70.50_{0.20}$	$\downarrow 69.34_{0.87}$
Arabic	$76.01_{0.61}$	$\uparrow 76.43_{0.36}$	$\uparrow 76.58_{1.42}$	–	$\uparrow \mathbf{78.53}_{1.26}$

achieved for Slovenian by using LOO intermediate training (ΔF1 +2.26); followed by Arabic with Slovenian intermediate training (ΔF1 +1.13), Croatian with Slovenian intermediate training (ΔF1 +1.02), and German with English intermediate training (ΔF1 +0.17). English performance decreases with all the intermediate training variants. Using cseBERT shows a similar trend, where the largest gain is for Arabic (ΔF1 +2.52), then Croatian (ΔF1 +1.56), Slovenian (ΔF1 +0.67) and English (ΔF1 +0.63), while performance for German decreases (ΔF1 −2.38). However, the gains using LOO (all available non-target language data) are always either the highest or very close to it, suggesting that this is the most useful practical approach in most cases. There is no conclusive evidence of the role played by the script; for example, Arabic intermediate training improves the performance of Croatian and Slovenian with mBERT while the performance decreases for English and German. Overall it seems that although intermediate training can provide gains, they are relatively small in most cases: whenever there is a large amount of data available for a task, training on the target task is likely to be sufficient to achieve optimal performance on that dataset, and using intermediate training in a different language(s) is unlikely to give significant gains.

Data hunger of the model

We next explore the effect of different amounts of training data, first in the monolingual, target-language-only case (Fig. 2), and then with intermediate training (Figs. 3 and 4).

Figure 2 shows the increasing data training regime without intermediate training, and shows a substantial difference between the performance with the mBERT and cseBERT LMs. With Croatian and Slovenian (the less-resourced languages on which cseBERT is

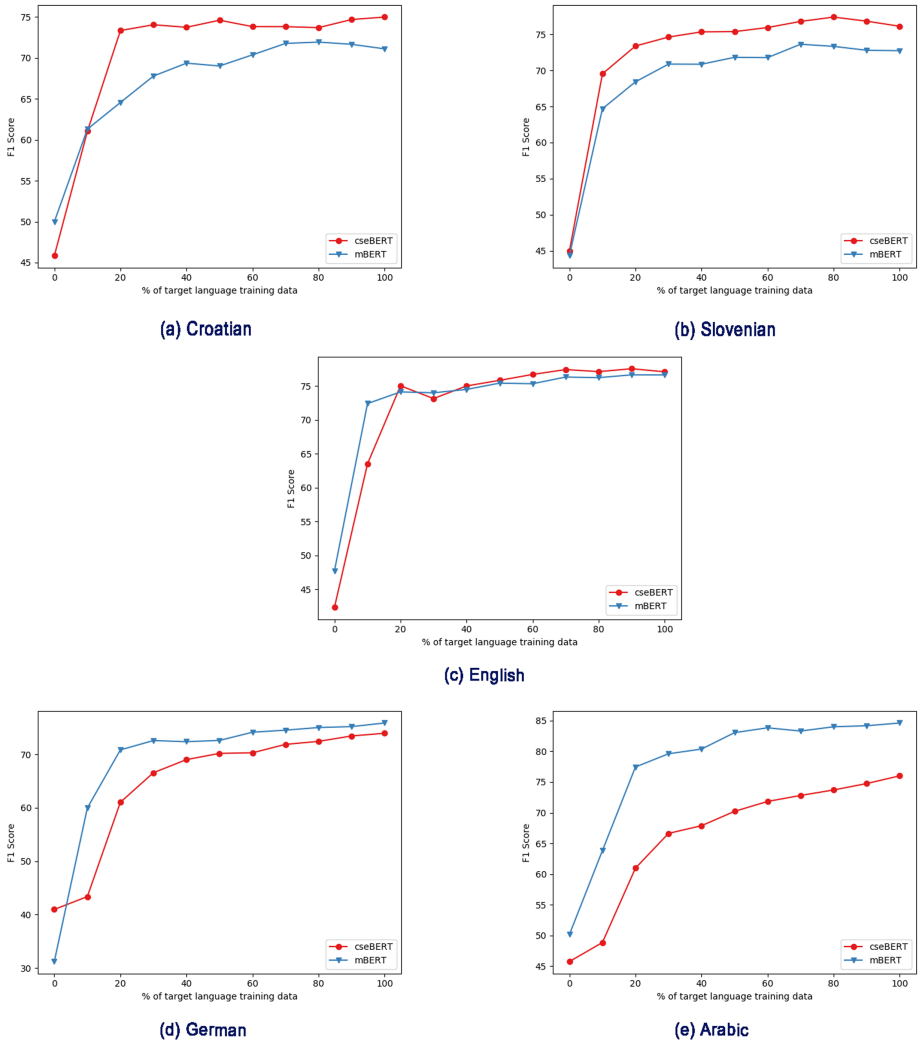


Figure 2 Effect of different pre-trained LMs (mBERT vs cseBERT), with varying amount of target language training data in the fine-tuning step, and no intermediate training. (A) Croatian, (B) Slovenian, (C) English, (D) German, (E) Arabic. Full-size DOI: 10.7717/peerj-cs.559/fig-2

trained), not only does cseBERT outperform mBERT (following the full-dataset results in Table 2), but performance is relatively high, and increase over mBERT is substantial, even with a very small amount of training data (e.g., 10%). On the other hand, for German and Arabic, mBERT outperforms cseBERT. For English, performance is similar, reconfirming the pattern from Table 2 that on English there is no large gain by using the cseBERT model.

Next, we apply the same regime of gradually increasing the amount of target-language fine-tuning data, but this time after using intermediate training (thus testing the scenario where we have large amounts of data in similar tasks in other languages but little in

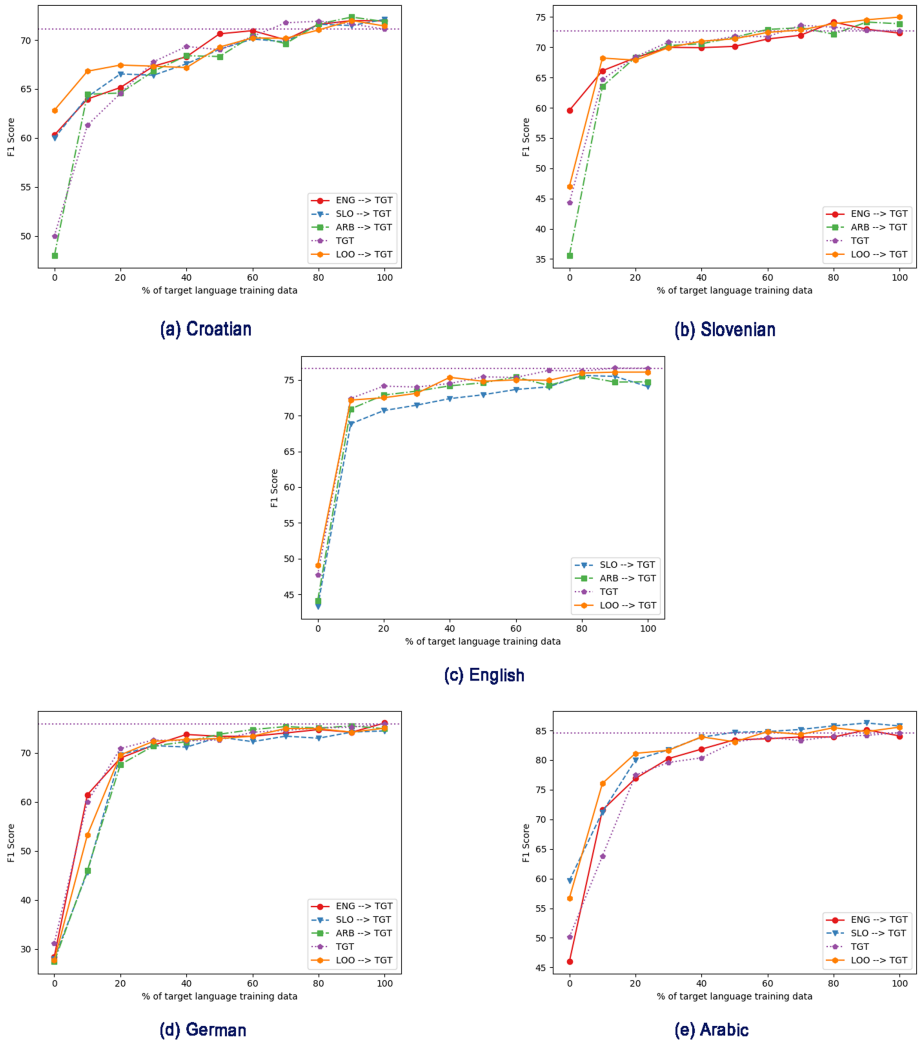


Figure 3 Effect of different intermediate training languages, with varying amount of target language training data in the fine-tuning step, using mBERT. TGT: Only fine-tuned on target language (no intermediate training). (A) Croatian, (B) Slovenian, (C) English, (D) German, (E) Arabic. ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO → TGT: Intermediate training on all non-target languages, then fine-tuning on target language.
Full-size DOI: 10.7717/peerj-cs.559/fig-3

the target language). Figures 3 and 4 show the results for mBERT and cseBERT respectively, including results without intermediate training, for comparison. In most cases, for comparatively low amounts of target-language data (~10%), intermediate training improves the results compared to fine-tuning purely on the target task if it is done using all the non-target languages available (see Table 5). In this case, we observe statistically significant improvements in 6 out of 10 experimental settings: for Slovenian and Croatian (with both LM), English (with cseBERT) and Arabic (with mBERT). For the

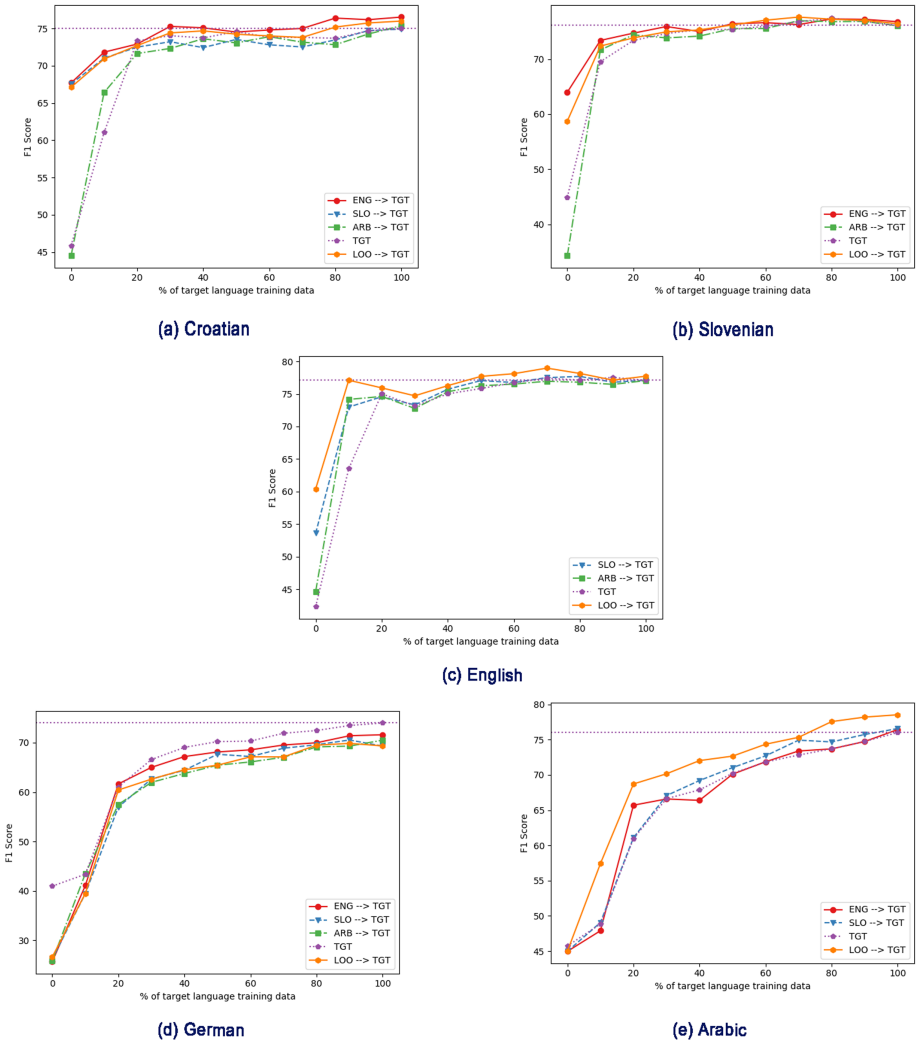


Figure 4 Effect of different intermediate training language with varying amount of target training data, using cseBERT. TGT: Only fine-tuned on target language (no intermediate training). (A) Croatian, (B) Slovenian, (C) English, (D) German, (E) Arabic. ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO → TGT: Intermediate training on all non-target languages, then fine-tuning on target language.

Full-size DOI: 10.7717/peerj-cs.559/fig-4

other 4 settings, the results slightly degrade but the differences are not statistically significant. For settings, when we used only one language for intermediate training, the results seem to be inconclusive.

However, when more target language data is available, the gains from intermediate training drop. In other words, intermediate training only helps when target-language data is scarce. We can also see that intermediate training does not always lead to improved performance (shown also in experiments in Table 4). For example, for Croatian, using

Table 5 Comparison of mBERT and cseBERT with intermediate training using all non-target languages (LOO setting) and fine-tuning on only 10% training data in the target language. Values are shown as macro-averaged F1-scores. Differences marked with † are statistically significant. Bold indicates the best performance for each language.

Language	mBERT		cseBERT	
	TGT	LOO → TGT@10%	TGT	LOO → TGT@10%
Croatian	61.30	†**66.82**	61.04	†**70.91**
Slovenian	64.68	†**68.22**	69.52	†72.63
English	**72.40**	72.17	63.51	†**77.11**
German	**59.97**	53.20	**43.36**	39.64
Arabic	63.82	†**76.07**	48.84	**57.42**

Table 6 Area Under the Curve (AUC) of F1-score as we vary amount of target language training data in the fine-tuning step from 0% to 100%, for different intermediate training languages. TGT: Only fine-tuned on target language (no intermediate training). ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO → TGT: Intermediate training on all non-target languages, then fine-tuning on target language. Bold indicates the best performance for each target language. Pairwise statistical tests for each training setting show statistically significant differences between mBERT and cseBERT results for all settings.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	mBERT				
Croatian	$67.82_{1.22}$	↑$68.61_{0.78}$	↑$68.28_{0.45}$	↓$67.66_{0.24}$	↑**68.86**$_{0.25}$
Slovenian	$69.67_{0.73}$	↑$70.09_{0.10}$	–	↓$69.16_{0.24}$	↑**70.32**$_{0.14}$
English	**73.71**$_{0.32}$	–	↓$71.38_{0.38}$	↓$72.52_{0.01}$	↓$73.25_{0.21}$
German	**70.10**$_{0.50}$	↓$69.76_{0.24}$	↓$67.51_{0.55}$	↓$68.27_{0.47}$	↓$68.95_{1.37}$
Arabic	$78.70_{0.16}$	↑$79.55_{0.26}$	↑$81.63_{0.47}$	–	↑**81.64**$_{0.09}$
	cseBERT				
Croatian	$71.31_{1.36}$	↑**74.42**$_{0.19}$	↑$72.75_{0.22}$	↓$71.12_{0.39}$	↑$73.73_{0.26}$
Slovenian	$73.57_{0.29}$	↑**75.31**$_{0.17}$	–	↓$73.08_{0.33}$	↑$74.91_{0.13}$
English	$73.10_{0.80}$	–	↑$74.78_{0.13}$	↑$74.08_{0.51}$	↑**76.32**$_{0.24}$
German	**65.59**$_{0.71}$	↓$63.11_{0.46}$	↓$61.51_{0.43}$	↓$61.19_{0.81}$	↓$61.40_{0.16}$
Arabic	$66.85_{0.94}$	↑$67.11_{0.37}$	↑$67.63_{0.77}$	–	↑**70.82**$_{0.85}$

intermediate training on mBERT with a large amount of data decreases performance, while with cseBERT the performance is consistently improved. For mBERT on English, using Slovenian data for intermediate training clearly decreases performance. For Slovenian and Arabic, performance improves in all intermediate training settings, even with the full amount of training data. For cseBERT and Arabic, we can see that the LOO setting brings important gains in the performance, which can be explained by the fact that the LOO setting contains training data in languages used in the cseBERT pre-training. For English and cseBERT, we can clearly see that the LOO intermediate training is very useful if we have less than 80% of target data available.

To quantify the overall gains, in Table 6 we report the area under the F1-score curve (AUC) as the target language dataset size varies from 0% to 100% (see Figs. 3 and 4).

Overall, we see that intermediate training helps; the exceptions are German for both mBERT and cseBERT, and English when using mBERT. The highest gain can be observed for Arabic and Croatian with cseBERT (improving by ~4% and ~3%, respectively); both languages show gains with mBERT too, although smaller. The gain in Arabic strongly suggests that intermediate training helps even if scripts are different. For Slovenian when using cseBERT we also gain more than ~1% with intermediate training on English, and when using mBERT less than ~1% with LOO setting. For German, performance is inconsistent: with English intermediate training, performance drops by ~1%, and with Slovenian it improves by ~1%.

In terms of cseBERT and mBERT comparison, the results are consistent with those in Table 2: cseBERT improves over mBERT for the languages it is trained on (Croatian and Slovenian). For Arabic there is a large performance gap (~11%) between mBERT and cseBERT. We hypothesize that this is due to vocabulary: the cseBERT model sees no Arabic words in pre-training. cseBERT also doesn't know German words, but the performance drop for German is much lower than for Arabic (less than ~5%); therefore we hypothesize that due to the Latin script of German and relative closeness to English and Slovenian, the sub-word tokenization provides some common vocabulary. German is closer to English as both are Germanic languages, but German also had a historically big influence on the evolution of the Slovenian language, therefore, there are bound to be words with similar roots.

With this quantitative analysis, we have shown that cross-lingual transfer can be effective for the offensive speech detection task, giving results with good performance even with small amounts of target language data. Using a better language-specific multilingual BERT (here, cseBERT) improves performance for languages that are less well represented in the standard mBERT model, and requires comparatively less target language data to achieve close to optimal performance. However, using different language task data as intermediate training doesn't improve the performance in all cases; but when the target-language dataset size is small, intermediate training does give improvements.

ANALYSIS AND QUALITATIVE RESULTS

In this section, we take a closer look at the performance of the models. In "Analysis of Misclassification", we examine how mBERT and cseBERT differ in their mistakes, with a per-example analysis of several trained models to explore how the misclassifications change with different pre-trained language models. In "Analysis of Classifier Confidence", we go further and examine misclassifications and different kinds of example via patterns in the confidence of the model outputs. While in "Analysis of Vocabulary Coverage", we look at the vocabulary coverage and compare it with the model's performance.

Analysis of misclassification

We analyze the performance of mBERT and cseBERT using misclassified examples, aiming to explore how the space of misclassified samples behaves and changes when we change the underlying language model. Although standard performance metrics give us some idea of the models' performance varies on different classes, they do not provide

any insight into the performance across particular examples. For example, two models may achieve the same overall accuracy score yet may misclassify completely different examples.

The analysis is performed on the three languages of cseBERT (Croatian, Slovenian and English); for each language, we perform a pair-wise comparison of mBERT and cseBERT model outputs. All compared models were trained using 100% of target language training data without any intermediate training (corresponding to the quantitative results in Table 2). Figure 5 presents, for each comparison, the percentage of misclassified test set examples in the form of Venn diagrams, one for 'offensive' examples and one for 'not offensive' (according to the gold-standard labels). The different subsets in the diagrams show the proportions misclassified by mBERT alone, by cseBERT alone, and by both models together.

Figures 5E and 5F show that mBERT and cseBERT perform similarly for English. The subset of examples misclassified by both models is relatively large, covering 58% of the offensive and 37% of not-offensive examples. The other two subsets are of similar size: each model corrected some mistakes from the other model but made a similar number of mistakes on other examples. The results seem to be more in favor of cseBERT for the Slovenian and Croatian languages (see Figs. 5A–5D). Fewer examples are misclassified by cseBERT than mBERT, except for the Croatian 'not offensive' case. For these two languages, the proportion of shared misclassified examples is also much lower than for English, in all settings except for the Croatian 'offensive' examples (56%), where it is close to (but still lower than) the 'offensive' English examples.

These results show that while cseBERT does not seem to have any advantage for English, it performs substantially better for Slovenian and Croatian, in line with the quantitative results of Table 2. For these languages, it correctly classifies a range of examples for which mBERT makes incorrect predictions. Furthermore, the reduced number of the Slovenian and Croatian shared misclassifications may suggest that these models have gained different knowledge during their pre-training phases. These results show great promise for using these two models in tandem, e.g., as part of an ensemble, to produce higher quality models for hate speech detection in Slovenian and Croatian.

Analysis of classifier confidence

In this section, we look for patterns in the outputs based on the classifier's confidence. Specifically, we analyze how "true" label confidence varies as the model is trained using more and more data (see data hunger analysis in "Quantitative Results"). Formally, for a test instance (x_i) on the j% of the target data at the kth epoch, we looked at the correct label probability for all trained models. The *confidence* of the classifier is defined as the mean of the correct label probabilities and the *variability* the standard deviation.
We analyzed the *confidence* and *variability* together to find the overall behavior of the test data Following *Swayamdipta et al. (2020)*, we plot *confidence* and *variability* on the Y-axis and X-axis respectively. Please note that *Swayamdipta et al. (2020)* calculated confidence and variability over epochs; we used both changes over the data size and epochs. Figure 6 shows the confidence-variability plot for the English data; we found a similar pattern for other languages. As we can see from Fig. 6, there are three groups of

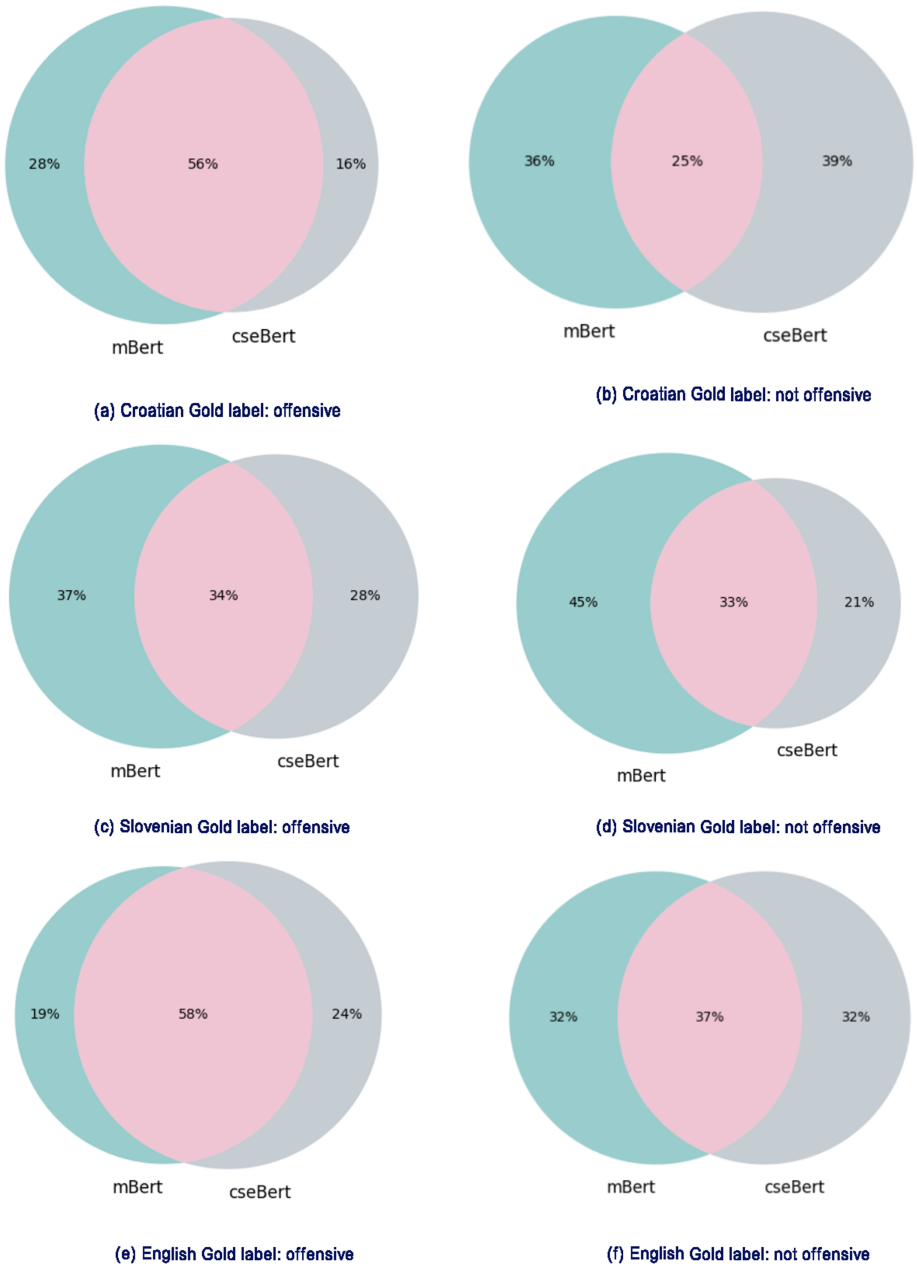


Figure 5 Comparison of misclassified examples for the mBERT and cseBERT models trained on 100%data with no intermediate learning step. (A) Croatian Gold label: offensive; (B) Croatian Gold label: not offensive; (C) Slovenian Gold label: offensive; (D) Slovenian Gold label: not offensive; (E) English Gold label: offensive; (F) English Gold label: not offensive. Figures on the left show misclassified examples with the 'offensive' gold label; on the right, misclassified examples with the 'not offensive' gold label. Green subsets: misclassified by mBERT but correctly classified by cseBERT. Grey subsets: misclassified by cseBERT but correctly classified by mBERT. Violet subsets: misclassified by both models. Full-size DOI: 10.7717/peerj-cs.559/fig-5

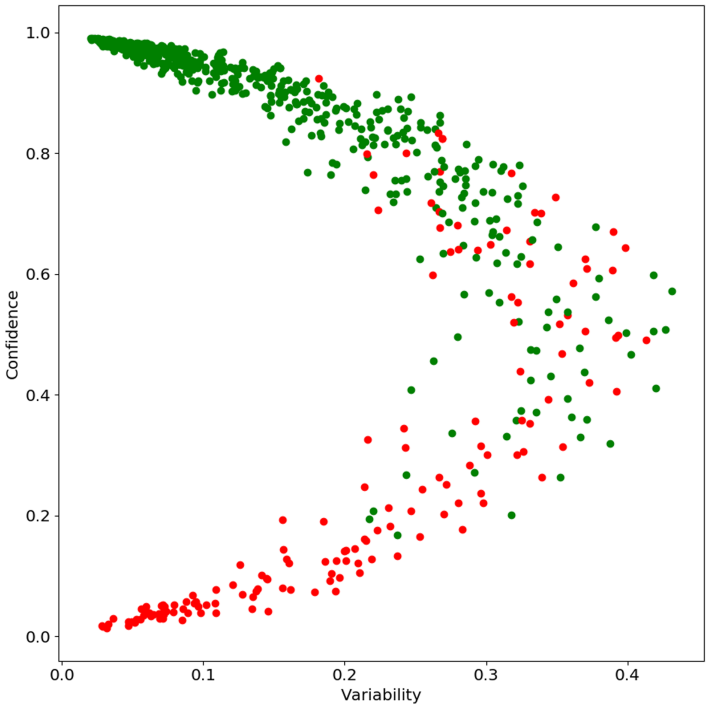


Figure 6 Confidence Score for English data: green when example is correct and red when example is incorrect by the best selected model with 100% data. Full-size DOI: 10.7717/peerj-cs.559/fig-6

instances. First, those for which the classifier is correct and has very high confidence and low variability, i.e., "easy" examples. Second, those where classifier confidence is close to 0.5 and has high variability, i.e., "ambiguous" examples. And third, where the classifier has very low confidence and variability for the true label, i.e., "hard" examples.

To further analyze these three categories, we manually inspected some examples and tried to understand what makes them easy, ambiguous, or hard for the classifier to classify. We present some of these examples in Tables 7–9. Most easy examples are characterized by specific offensive words or phrases. For example, in Table 7, the first example has "Nigga ware da", and the second example has only socially accepted words. In the hard category, many examples are cases where it is hard to identify from the sentence alone whether it is offensive or not, without some form of context. The classifier generally made mistakes in classifying such instances. For example, in Table 7, one example needs context in the form of the URL, and the other one is dependent on the comment it is replying to. The ambiguous category is perhaps the most interesting: in many cases, the annotation appears to be wrong, and in others another label is equally possible. For such examples, we have provided the potentially correct labels in the tables. The classifier seems to work inconsistently for these instances; we believe this is because these instances have patterns similar to the class opposite to their gold label. Please note that these three classes are not rigidly defined: several examples could belong to

Table 7 English examples from the test set belonging to different regions in the confidence-variability plot with gold labels and classifier labels. Examples which seem to have been assigned an incorrect gold label are given a possible correct label assigned based on our judgement.

Nr.	Category	Example	Gold label	Classifier label	Possible label	Remark
1	Easy	@USER Nigga ware da hits at	OFF	OFF	/	
2	Easy	#I once read about the man who invented bulbs, he saw failure differently, it was 5,000 new ways of doing it differently…Now he is a genius#	NOT	NOT	/	
3	Hard	#ChristineBlaseyFord is your #Kavanaugh accuser… #Liberals try this EVERY time… #ConfirmJudgeKavanaugh URL	OFF	NOT	/	Lack of context—from URL
	Hard	@USER @USER @USER Are you referring to how they lollygag with gun control as their kids get slaughtered in schools?	NOT	OFF	/	Lack of context
5	Ambiguous	#Kavanaugh so a wild claim from 36 years ago of groping has evolved into A Rape A Violent Sexual Event by Move URL a Soros based Org. that supports BLM Antifa etc. Unbelievable!	NOT	OFF	OFF	Gold label mistake. Offensive towards BLM etc.
6	Ambiguous	#Cancer Hate crime isn't an actual crime. If there's a victim, then it's just called crime. (Malum in se) "Hate" crime is socially engineered programing to control people's behavior, just like gun control. #ChangeMyMind URL*	NOT	OFF	BOTH	Both labels are possible. Giving opinion but could also be treated as pointing to blame to a gr. of people

Table 8 Slovenian examples from the test set belonging to different regions in the confidence-variability plot with gold labels and classifier labels. Examples which seem to have been assigned an incorrect gold label are given a possible correct label assigned based on our judgement.

Nr.	Category	Example	Gold Label	Classifier label	Possible Label	Remark
1	Easy	Postreljati to smrdljivo golazen pa direktno na grmado pa skurit [*Shoot this stinky vermin and burn it at the stake*]	OFF	OFF	/	
2	Easy	Been here, seen that :)	NOT	NOT	/	Different language
3	Hard	Na imigrantski mladini svet "stoji". https://www.youtube.com/watch?v=Wj9LLC7GZQk Pridruži se, če ti ni vseeno za svojo domovino: https://www.facebook.com/stranka.slovenskega.naroda.ssn [*The world depends on young migrants. Join if you care about your country.*]	NOT	OFF	/	Lack of context—from URL
4	Hard	V zivalski vrt jh iskat pa bo zadeva resena :) [*Go to the zoo and get them, problem solved :)*]	NOT	OFF	/	Lack of context
5	Ambiguous	Sej bo ze drzava placala ne skrb haha [*Don't worry, the government will pay haha*]	OFF	NOT	/	Lack of context
6	Ambiguous	Ce si rojen v sloveniji, to ne pomeni tud da si!!!!!!!!!!!!!vazne so korenine!!!!!!!! [*If you're born in Slovenia it doesn't mean you are a Slovenian!!!!!! Your roots matter!!!!!!*]	NOT	NOT	OFF	Gold label mistake

other classes. In particular, there are overlaps between the hard and ambiguous classes: in many cases the gold labels appear to be wrong for "hard" examples, and "ambiguous" examples require context. However, most such overlaps occur at the boundaries of the classes.

Table 9 Croatian examples from the test set belonging to different regions in the data map with gold labels and classifier labels. Examples which seem to have been assigned an incorrect gold label have a possible correct label assigned based on our judgment.

Nr.	Category	Example	Gold label	Classifier label	Possible label	Remark
	Easy	Ja san dobia zuti karton jer san covika oslovia sa klaune a to sto oni reklamiraju javno prostituciju, lazi, itd nikome nista… Admini ove stranice naguzite se mamicu [*I got a warning because I said to someone that he was a clown but they are advertising public prostitution, spreading lies etc. and nothing happens… Admins of this site are motherfuckers.*]	OFF	OFF	/	
2	Easy	Ko si ti kurvo glupa da nekome nešto govoris [*Who are you stupid whore to lecture someone*]	OFF	OFF	/	
3	Hard	Treba iz objesiti ! [*Needs to be hanged!*]	OFF	NOT	/	Lack of context
4	Hard	Gospođo, u kuhinju! [*Go to the kitchen, miss!*]	OFF	NOT	/	Sociolinguistic features
5	Ambiguous	Vaso jedi kurac [*Vaso eat dick*]	NOT	OFF	OFF	Gold label mistake
6	Ambiguous	Da je pravde po mom na ovom svijetu završile bi njemu ruke na giljotini pa nek boksa ćaću svog… Dizat ruku na Policiju ma mrs tamo [*If there were justice in this world his hands would end up on a guillotine and then he could start hitting his father… Striking a policeman, what the hell*]	NOT	OFF	OFF	Gold label mistake

For the Slovenian dataset, we found some examples written in a language other than Slovenian (see example 2 Table 8). We observe that on average such instances tend to get correctly classified, perhaps due to the effectiveness of the multilingual mBERT and cseBERT representations, or because the English used in these cases is relatively simple; however, no conclusions can be made without deeper analysis.

For Slovenian and Croatian, another category of examples was found that cannot be labeled without more general cultural and societal knowledge. We currently do not know how much such knowledge, if any, a language model possesses, which may lead to difficulties in labeling such messages. A clear-cut example would be "Gospodo, u kuhinju!" (Go to the kitchen, miss!) from the Croatian dataset (see Table 9). Such an example may seem very tame in terms of its vocabulary; however, in gender roles, it may be labeled as offensive to women. Such examples can be found in any region (easy, hard or ambiguous) of the data map. This suggests the classifier seems to pick some signals for these kinds of instances during training, however, the results are highly inconsistent. In order for the classifier to classify such instances correctly, it seems likely that similar instances must be present in the training set during fine-tuning; the knowledge from the pre-trained model may not be enough to decode such instances properly.

Attention visualization

In Fig. 7 we provide an attention weight visualization for two English examples, one from the high-confidence/low-variability region (i.e., "easy") and another from the low-confidence/low-variability region of the data map ("hard"). For each instance we have visualized the maximum attention weight each token gets across BERT's 12 attention heads, using the AttViz visualization tool (*Škrlj et al., 2021*). Since the role of attention is to

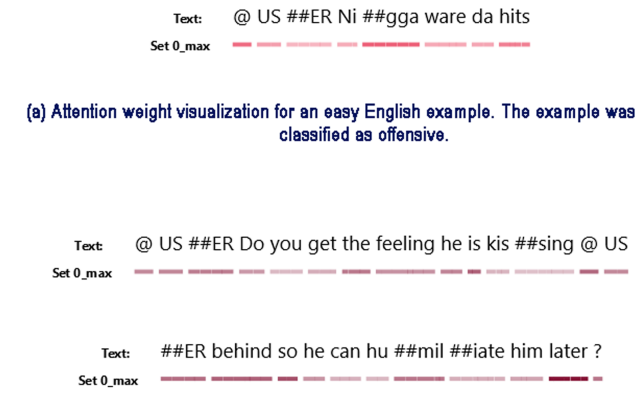


Figure 7 Attention weight comparison for an easy (A) and a hard (B) example in English.
Full-size DOI: 10.7717/peerj-cs.559/fig-7

weight different parts of the input, this lets us gauge the relative importance of specific input tokens.

As is standard with BERT models, we add two special tokens to the original input text during training and inference stages (see *Devlin et al., 2019*). The (CLS) token is added in the first position in the sequence, and its representation is used for performing classification. The (SEP) token is added in the last position of the input text sequence to mark its end. Since these two tokens are present in every input at predefined positions they are assigned high attention weights by the model. However, we are more interested in the importance of other tokens that are originally part of the input text. Since the presence of these two tokens during visualization may overshadow the importance of other tokens, we remove them from the input during visualization of the attention weights.

Figure 7A presents an "easy" example which was correctly classified by the model as offensive. We can see that the model puts a lot of weight on the token "##gga", part of the offensive word "nigga". It also puts moderate weight on the final word "hits" which may suggest violence. Figure 7B presents a "hard" English example. Here the model puts weight on the token "behind", however it is unable to decipher the meaning of the English expression "kissing someone's behind" and misclassifies the example as not offensive.

Analysis of vocabulary coverage

In this section, we shed some light on the performance difference based on vocabulary coverage. Specifically, we are interested in understanding whether better vocabulary coverage helps classification performance. To measure this, we calculated the percentage of missing words in the sentence, i.e., the words that are not present either in the pre-trained LM vocabulary or in the training set. BERT-based models use WordPiece (*Schuster & Nakajima, 2012*; *Wu et al., 2016*) to create the vocabulary. WordPiece is a data-driven approach guaranteed to generate a deterministic segmentation of a word. For example,

if "bagpipe" is not present in the vocabulary, but "bag" and "pipe" are, then "bagpipe" will be divided into two sub-words "bag" and "##pipe", where "##" indicates that a token is part of the previous word. This allows for wider vocabulary coverage, as even rare words can be covered via their sub-word units. We define a missing word as either:

- a word *split to character level* (and therefore not in the pre-trained model's vocabulary, although it may be present in the training data). The hypothesis behind this condition is that if words are split into individual characters rather than longer tokens, it is unlikely that a model can easily assign meaning.

  or

- a word *not in the vocabulary nor in the training set*. In this case, a word may be split into larger units than characters. If the word is present in the training set, it is not considered as missing: the meaning may at least partly be learned by the classifier model during the training phase.

We illustrate this with an example sentence "I like flowers", assuming that only "I" is present in the vocabulary, but "like" and "flowers" are present in the training set. If the sentence is tokenized as "I li ##ke flower ##s", then there are 0 missing words. However, if tokenized as "I l ##i ##k ##e flower ##s" (i.e., "like" is character-level tokenized), there is one missing word, i.e., 33.33%.

In Fig. 8, we plot the classifier F1 score against the cumulative percentage of missing words (i.e., for data with x% or less missing words, what is the performance). We also report the percentage of test set examples covered at that point. As we can see from Fig. 8, as the percentage of missing words increases, the performance decreases in most cases. There are a few exceptions: for Croatian, due to a sharp drop at 10% there is a large subsequent increase in performance. This could be due to more hard examples in that range.

For Croatian and Slovenian, cseBERT has fewer missing words than mBERT, and this better vocabulary coverage may be one reason for the performance gain. As we can see from Figures 8A and 8B, when there is less than 20% of missing words, cseBERT covers 3–5% more sentences for Croatian and Slovenian compared to mBERT, and shows a corresponding performance gain of more than 5–6%. However, this cannot be the only factor: at 0% missing words, even though there is only 1% higher dataset coverage, there is a large difference (4–5%) in performance. This could be due to larger whole-word vocabulary coverage, allowing cseBERT to learn better word meaning.

Interestingly for English (Fig. 8C), even though cseBERT has less vocabulary coverage, it performs slightly better. However, for German, the trend is the opposite: mBERT has less vocabulary coverage, and performs better, because it is pre-trained on the German data, while cseBERT is not. For Arabic, cseBERT has a very high percentage of missing words, with all the examples having more than 50% missing words (see Fig. 8E), and the difference between the cseBERT and mBERT performance is very high (11%, see Table 2).[4] Our results therefore show some links between vocabulary coverage and

[4] Please note that even though cseBERT is not trained on the Arabic script, it has some Arabic characters in the vocabulary and the Arabic dataset has some Latin words.

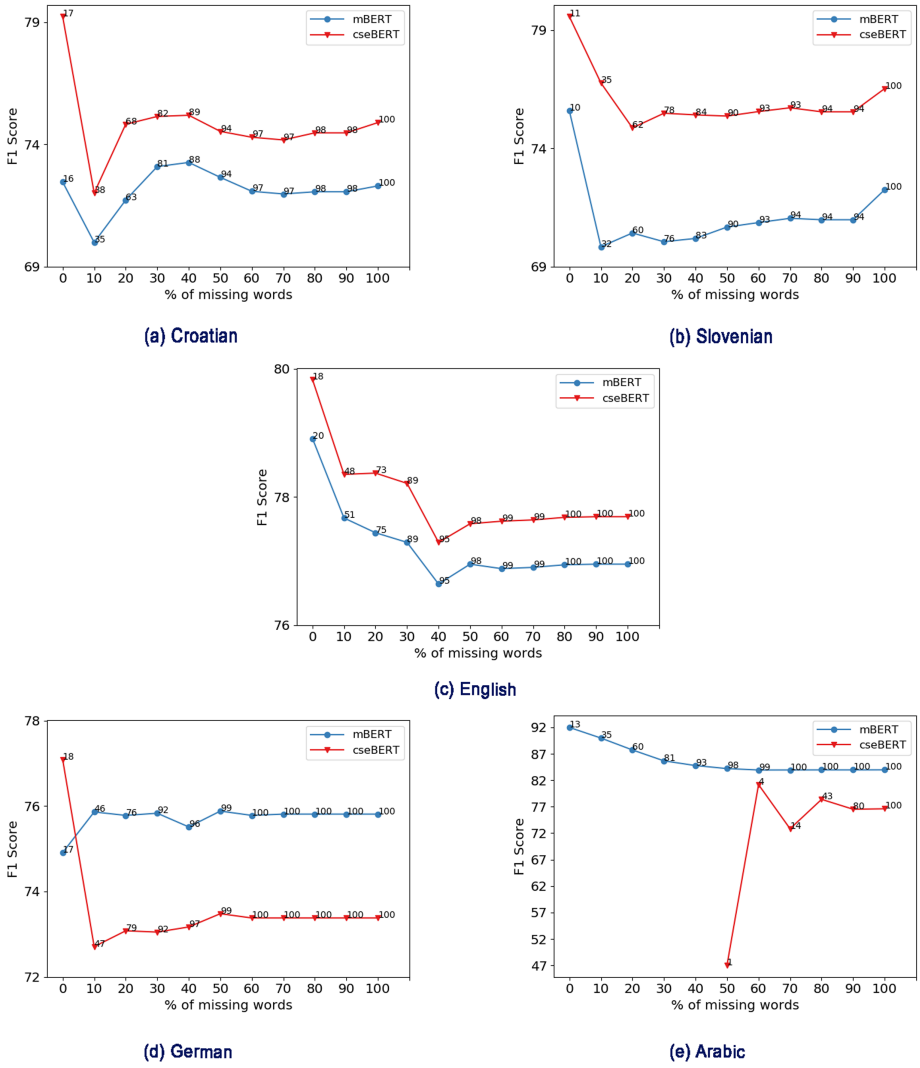


Figure 8 Effect of % of missing words (e.g., 30% means 30% or less missing words) on performance for mBERT and cseBERT. (A) Croatian, (B) Slovenian, (C) English, (D) German, (E) Arabic. Numbers on the lines represent % of test set samples covered at that point.

Full-size DOI: 10.7717/peerj-cs.559/fig-8

performance, but suggest that more research is needed to fully understand them. In the future, we plan to look at how these effects relate to word frequency and part of speech.

CONCLUSION

In this work, we study the feasibility of cross-lingual training to develop offensive speech detection models. Specifically, we investigated how the choice of pre-trained multilingual language models and non-target language intermediate training impact the final performance. We experimented with five diverse languages; Croatian, Slovenian,

English, German, and Arabic, using two pre-trained language models, mBERT and cseBERT. We found out that having a language model pre-trained with a smaller set of languages has a better overall performance than a general multilingual language model for those languages, and gives better performance via intermediate training. In general, intermediate training is not useful if a large amount of target language data is available, giving relatively small improvements in only approximately half of the experiments, regardless of choice of language or number of languages for intermediate training. However, intermediate training is useful when we have limited target language data, and is particularly effective with a good choice of pre-trained language model. In this case, intermediate training with all other available languages (LOO) boosted performance for all languages except German.

Considering the choice of language model had the most significant impact on the final model performance, we also performed a qualitative analysis of the two language models we used in this study, namely mBERT and cseBERT. Vocabulary analysis suggests that better vocabulary coverage could be one reason for better performance, but that it is probably not the only factor. The analysis using classifier confidence revealed that models generally have trouble classifying instances that are hard to understand without additional context. Furthermore, the models perform inconsistently where additional socio-political knowledge is required to label the message correctly.

In future work on cross-lingual hate speech detection, we would like to make our analysis more general by extending it to other languages and other NLP tasks, and extend our study to other multilingual language models beyond the BERT architecture, such as those based on XLM (*Conneau & Lample, 2019*).

APPENDIX

We present additional metrics to better gauge the performance of our models in various experimental settings conducted in the course of this study.

Tables 10 and 11 show the results of mBERT and cseBERT models respectively in terms of macro-averaged recall and precision when they are trained on all available target language data without intermediate training. For comparison with the F1 score, refer to the Table 2.

Tables 12 and 13 show the results of mBERT and cseBERT models respectively when intermediate training is performed in one or more non-target languages and no fine-tuning is performed on target language data (zero-shot setting). The performance of the models is measured in terms of macro-averaged recall and macro-averaged precision scores. For comparison with the F1 score, refer to Table 3.

Tables 14 and 15 show the results of mBERT and cseBERT models respectively when intermediate training is performed in one or more non-target languages and fine-tuning is performed on all available target language data. The performance of the models is measured in terms of macro-averaged recall and macro-averaged precision scores. For comparison with the F1 score, refer to Table 4.

The additional metrics seem to confirm our claims of model comparison between mBERT and cseBERT models. Both in scenarios where high amounts of target language

Table 10 Results for mBERT models, fine-tuned on all training data in the target language only (no intermediate training). Values are shown as recall and precision scores with standard deviation. Bold indicates the best performance for each language.

Language	Recall		Precision	
	Random init.	Fine-tuned	Random init.	Fine-tuned
Croatian	$51.68_{1.9}$	$\mathbf{69.14}_{1.3}$	$51.56_{1.8}$	$\mathbf{74.70}_{1.6}$
Slovenian	$49.30_{19.0}$	$\mathbf{72.67}_{0.4}$	$47.82_{3.7}$	$\mathbf{72.87}_{0.4}$
English	$51.70_{1.9}$	$\mathbf{75.89}_{1.1}$	$52.14_{1.6}$	$\mathbf{77.56}_{1.3}$
German	$49.47_{0.5}$	$\mathbf{75.16}_{0.3}$	$48.33_{0.4}$	$\mathbf{77.14}_{0.6}$
Arabic	$49.13_{1.5}$	$\mathbf{83.48}_{0.6}$	$48.39_{1.8}$	$\mathbf{85.98}_{0.4}$

Table 11 Results for cseBERT models, fine-tuned on all training data in the target language only (no intermediate training). Values are shown as recall and precision scores with standard deviation. Bold indicates the best performance for each language.

Language	Recall		Precision	
	Random init.	Fine-tuned	Random init.	Fine-tuned
Croatian	$48.34_{2.2}$	$\mathbf{73.38}_{0.9}$	$48.77_{1.6}$	$\mathbf{77.33}_{1.5}$
Slovenian	$48.96_{1.6}$	$\mathbf{76.17}_{0.5}$	$49.15_{1.9}$	$\mathbf{76.11}_{0.6}$
English	$50.91_{1.2}$	$\mathbf{76.46}_{1.2}$	$50.87_{0.9}$	$\mathbf{77.88}_{1.5}$
German	$50.90_{2.4}$	$\mathbf{73.38}_{1.1}$	$56.70_{7.9}$	$\mathbf{74.96}_{0.9}$
Arabic	$51.48_{4.4}$	$\mathbf{74.32}_{0.9}$	$50.94_{3.6}$	$\mathbf{78.46}_{1.2}$

Table 12 Results of intermediate training in a range of non-target languages in zero-shot transfer on the target language data for mBERT models using macro-averaged recall (top) and macro-averaged precision (bottom) scores. TGT: random initialization (no intermediate training, no target fine-tuning). ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then zero-shot transfer on the target language. LOO → TGT: Intermediate training on all non-target languages, then zero-shot transfer on the target language. Bold indicates the best performance for each language.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	Recall				
Croatian	$51.68_{1.9}$	$\uparrow 55.66_{0.0}$	$\uparrow \mathbf{65.96}_{0.0}$	$\downarrow 50.44_{0.0}$	$\uparrow 65.48_{0.0}$
Slovenian	$49.30_{19.0}$	$\uparrow 53.25_{0.0}$	–	$\uparrow 51.69_{0.0}$	$\uparrow \mathbf{56.16}_{0.0}$
English	$51.70_{1.9}$	–	$\downarrow 51.34_{0.0}$	$\downarrow 50.73_{0.0}$	$\uparrow \mathbf{54.21}_{0.0}$
German	$\mathbf{49.47}_{0.5}$	$\downarrow 46.76_{0.0}$	$\downarrow 45.76_{0.0}$	$\downarrow 47.33_{0.0}$	$\downarrow 41.70_{0.0}$
Arabic	$49.13_{1.5}$	$\uparrow 50.31_{0.0}$	$\uparrow \mathbf{56.80}_{0.0}$	–	$\uparrow 55.40_{0.0}$
	Precision				
Croatian	$51.56_{1.8}$	$\uparrow \mathbf{65.85}_{0.0}$	$\uparrow 61.96_{0.0}$	$\uparrow 51.76_{0.0}$	$\uparrow 62.47_{0.0}$
Slovenian	$47.82_{3.7}$	$\uparrow 62.82_{0.0}$	–	$\uparrow 64.51_{0.5}$	$\uparrow \mathbf{65.70}_{0.0}$
English	$52.14_{1.6}$	–	$\uparrow 69.65_{0.0}$	$\uparrow 52.31_{0.0}$	$\uparrow 61.96_{0.0}$
German	$\mathbf{48.33}_{0.4}$	$\downarrow 38.32_{0.0}$	$\downarrow 39.83_{0.0}$	$\downarrow 43.32_{0.0}$	$\downarrow 32.93_{0.0}$
Arabic	$49.93_{1.8}$	$\uparrow \mathbf{89.85}_{0.0}$	$\uparrow 64.14_{0.0}$	–	$\uparrow 62.41_{0.0}$

Table 13 Results of intermediate training in a range of non-target languages in zero-shot transfer on the target language data for cseBERT models using macro-averaged recall (top) and macro-averaged precision (bottom) scores. TGT: random initialization (no intermediate training, no target fine-tuning). ENG/SLO/AR →TGT: Intermediate training on English/Slovenian/Arabic, then zero-shot transfer on the target language. LOO → TGT: Intermediate training on all non-target languages, then zero-shot transfer on the target language. Bold indicates the best performance for each language.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	Recall				
Croatian	$48.34_{2.2}$	↑$\mathbf{72.87}_{0.0}$	↑$70.19_{0.0}$	↑$49.51_{0.0}$	↑$71.97_{0.0}$
Slovenian	$48.96_{1.6}$	↑$\mathbf{66.81}_{0.0}$	–	↓$49.79_{0.0}$	↑$60.13_{0.0}$
English	$50.91_{1.2}$	–	↑$58.13_{0.0}$	↓$49.84_{0.0}$	↑$\mathbf{61.26}_{0.0}$
German	$\mathbf{50.90}_{2.4}$	↓$49.38_{0.0}$	↓$50.11_{0.0}$	↓$50.54_{0.0}$	↓$50.10_{0.0}$
Arabic	$\mathbf{51.48}_{4.4}$	↓$50.31_{0.0}$	↓$50.31_{0.0}$	–	↓$50.63_{0.0}$
	Precision				
Croatian	$48.77_{1.6}$	↑$\mathbf{67.63}_{0.0}$	↑$67.34_{0.0}$	↓$38.75_{0.0}$	↑$66.62_{0.0}$
Slovenian	$49.15_{1.9}$	↑$\mathbf{69.52}_{0.0}$	–	↑$45.45_{0.0}$	↑$68.22_{0.0}$
English	$50.87_{0.9}$	–	↑$73.75_{0.0}$	↓$36.01_{0.0}$	↑$\mathbf{77.15}_{0.0}$
German	$56.70_{7.9}$	↓$36.02_{0.0}$	↓$54.94_{0.0}$	↓$55.77_{0.0}$	↑$\mathbf{67.43}_{0.0}$
Arabic	$50.94_{3.6}$	↑$89.85_{0.0}$	↑$89.85_{0.0}$	–	↑$\mathbf{89.90}_{0.0}$

Table 14 Results of intermediate training in a range of non-target languages, followed by fine-tuning on all target language data for mBERT models using macro-averaged recall (top) and macro-averaged precision (bottom) scores. TGT: Only fine-tuned on target language (no intermediate training). ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO → TGT: Intermediate training on all non-target languages, then fine-tuning on target language. Bold indicates the best performance for each language.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	Recall				
Croatian	$69.14_{1.3}$	↑$70.06_{1.6}$	↑$\mathbf{70.14}_{0.4}$	↑$69.92_{0.8}$	↑$69.57_{0.3}$
Slovenian	$72.67_{0.4}$	↓$72.26_{1.1}$	–	↑$73.83_{0.7}$	↑$\mathbf{74.95}_{1.0}$
English	$\mathbf{75.89}_{1.1}$	–	↓$73.18_{0.6}$	↓$73.92_{0.6}$	↓$75.25_{0.6}$
German	$75.16_{0.3}$	↑$\mathbf{75.25}_{0.2}$	↓$73.89_{0.1}$	↓$74.21_{1.2}$	↓$74.23_{0.6}$
Arabic	$83.48_{0.6}$	↓$82.83_{1.1}$	↑$84.55_{1.3}$	–	↑$\mathbf{84.06}_{0.6}$
	Precision				
Croatian	$74.70_{1.6}$	↑$75.35_{1.6}$	↑$\mathbf{75.58}_{0.8}$	↑$75.41_{1.4}$	↑$74.85_{1.5}$
Slovenian	$72.87_{0.4}$	↓$72.75_{0.9}$	–	↑$74.03_{0.6}$	↑$\mathbf{75.10}_{1.2}$
English	$\mathbf{77.56}_{1.3}$	–	↓$75.33_{1.8}$	↓$75.83_{0.3}$	↓$77.20_{0.9}$
German	$77.14_{0.6}$	↑$\mathbf{77.46}_{0.4}$	↓$75.30_{0.0}$	↓$76.06_{1.0}$	↓$76.40_{0.2}$
Arabic	$85.98_{0.4}$	↓$85.61_{0.5}$	↑$87.16_{0.6}$	–	↑$\mathbf{87.37}_{0.5}$

data are available and in scenarios where target language data is not available (zero-shot scenario), the cseBERT consistently shows higher performance than mBERT on Croatian, Slovenian and English languages.

Table 15 Results of intermediate training in a range of non-target languages, followed by fine-tuning on all target language data for cseBERT models using macro-averaged recall (top) and macro-averaged precision (bottom) scores. TGT: Only fine-tuned on target language (no intermediate training). ENG/SLO/AR → TGT: Intermediate training on English/Slovenian/Arabic, then fine-tuning on target language. LOO → TGT: Intermediate training on all non-target languages, then fine-tuning on target language. Bold indicates the best performance for each language.

Target	TGT	ENG → TGT	SLO → TGT	AR → TGT	LOO → TGT
	Recall				
Croatian	$73.38_{0.9}$	$\uparrow 74.66_{1.2}$	$\downarrow 73.35_{0.5}$	$\downarrow 73.21_{0.5}$	$\uparrow \mathbf{74.67}_{0.8}$
Slovenian	$76.17_{0.5}$	$\uparrow \mathbf{76.76}_{0.4}$	–	$\downarrow 76.10_{0.5}$	$\uparrow 76.48_{0.3}$
English	$76.46_{1.2}$	–	$\downarrow 76.25_{0.8}$	$\downarrow 76.17_{1.2}$	$\uparrow \mathbf{76.70}_{0.5}$
German	$\mathbf{73.38}_{1.1}$	$\downarrow 70.85_{1.1}$	$\downarrow 68.37_{0.4}$	$\downarrow 69.88_{0.3}$	$\downarrow 68.69_{0.8}$
Arabic	$74.32_{0.9}$	$\uparrow 75.09_{0.5}$	$\uparrow 74.89_{1.3}$	–	$\uparrow \mathbf{76.72}_{1.4}$
	Precision				
Croatian	$77.33_{1.5}$	$\uparrow \mathbf{79.41}_{0.9}$	$\downarrow 77.26_{1.0}$	$\uparrow 78.93_{1.1}$	$\uparrow 77.80_{0.4}$
Slovenian	$76.11_{0.6}$	$\uparrow \mathbf{76.83}_{0.3}$	–	$\downarrow 76.05_{0.5}$	$\uparrow 76.40_{0.3}$
English	$77.88_{1.5}$	–	$\uparrow 78.26_{1.0}$	$\uparrow 78.25_{0.6}$	$\uparrow \mathbf{79.11}_{0.1}$
German	$\mathbf{74.96}_{0.9}$	$\downarrow 73.10_{0.8}$	$\downarrow 72.10_{0.5}$	$\downarrow 71.66_{0.2}$	$\downarrow 70.67_{1.0}$
Arabic	$78.46_{1.2}$	$\downarrow 78.18_{0.7}$	$\uparrow 78.97_{2.0}$	–	$\uparrow \mathbf{81.08}_{1.6}$

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This research is supported by the European Union's Horizon 2020 research and innovation program under Grant Agreement No. 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). The results of this publication reflect only the authors' views, and the Commission is not responsible for any use that may be made of the information it contains. Andraž Pelicon was funded also by the European Union's Rights, Equality and Citizenship Program (2014–2020) project IMSyPP (Innovative Monitoring Systems and Prevention Policies of Online Hate Speech, Grant No. 875263). Matthew Purver is also supported by the EPSRC under grant EP/S033564/1. This work is also supported by the Slovenian Research Agency (ARRS) core research program Knowledge Technologies (P2-0103), the research project CANDAS - Computer-assisted multilingual news discourse analysis with contextual embeddings (Grant no. J6-2581) and the young researchers' program for the work of Blaž Škrlj. There was no additional external funding received for this study. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
European Union's Horizon: 825153.
European Union's Rights, Equality and Citizenship Program: 875263.

EPSRC: EP/S033564/1.
Slovenian Research Agency (ARRS): P2-0103.
Slovenian Research Agency (ARRS): J6-2581.

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Andraž Pelicon conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Ravi Shekhar conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Blaž Škrlj conceived and designed the experiments, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Matthew Purver conceived and designed the experiments, authored or reviewed drafts of the paper, and approved the final draft.
- Senja Pollak conceived and designed the experiments, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The SemEval corpora in English, German and Arabic datasets are available under the Creative Commons Attribution 4.0 International License, and the training/evaluation/test splits for exact reproduction of our experiments are available at: https://github.com/EMBEDDIA/cross-lingual_training_for_offensive_language_detection.

The Slovenian data splits that we used in our experiments were provided for peer-review; the code used to split the Slovenian dataset is available at GitHub (in the module data_prep.py): https://github.com/EMBEDDIA/cross-lingual_training_for_offensive_language_detection.

For Croatian (24sata), the data is available as part of the EMBEDDIA project and is available at: https://www.clarin.si/repository/xmlui/handle/11356/1399.

Supplemental Information

Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj-cs.559#supplemental-information.

REFERENCES

Artetxe M, Schwenk H. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics* 7:597–610 DOI 10.1162/tacl_a_00288.

Bai X, Merenda F, Zaghi C, Caselli T, Nissim M. 2018. RuG@ EVALITA 2018: hate speech detection in Italian social media. In: *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018). Available at http://ceur-ws.org/Vol-2263/paper042.pdf*.

Basile A, Rubagotti C. 2018. CrotoneMilano for AMI at Evalita2018: a performant, cross-lingual misogyny detection system. In: *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018). Available at http://ceur-ws.org/Vol-2263/paper034.pdf*.

Basile V, Bosco C, Fersini E, Debora N, Patti V, Pardo FMR, Rosso P, Sanguinetti M. 2019. SemEval-2019 task 5: multilingual detection of hate speech against immigrants and women in Twitter. In: *Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational Linguistics*. 54–63.

Beyrer C, Kamarulzaman A. 2017. Ethnic cleansing in Myanmar: the Rohingya crisis and human rights. *The Lancet* **390(10102)**:1570–1573 DOI 10.1016/S0140-6736(17)32519-9.

Blair T. 2019. Designating hate: new policy responses to stop hate crime. *Available at https://institute.global/policy/designating-hate-new-policy-responses-stop-hate-crime*.

Chopra S, Sawhney R, Mathur P, Shah RR. 2020. Hindi–English hate speech detection: author profiling, debiasing, and practical perspectives. *Proceedings of the AAAI Conference on Artificial Intelligence* **34(01)**:386–393 DOI 10.1609/aaai.v34i01.5374.

Conneau A, Khandelwal K, Goyal N, Chaudhary V, Wenzek G, Guzmán F, Grave E, Ott M, Zettlemoyer L, Stoyanov V. 2020. Unsupervised cross-lingual representation learning at scale. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics*. 8440–8451.

Conneau A, Lample G. 2019. Cross-lingual language model pretraining. In: *Advances in Neural Information Processing Systems 32 (Proceedings of NeurIPS 2019). Available at https://proceedings.neurips.cc/paper/2019/file/c04c19c2c2474dbf5f7ac4372c5b9af1-Paper.pdf*.

Davidson T, Warmsley D, Macy M, Weber I. 2017. Automated hate speech detection and the problem of offensive language. In: *Eleventh International AAAI Conference on Web and Social Media*. 512–515.

Devlin J, Chang M-W, Lee K, Toutanova K. 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 (Long and Short Papers)*. Vol. 1. Minneapolis: Association for Computational Linguistics, 4171–4186.

Farha IA, Magdy W. 2020. Multitask learning for Arabic offensive language and hate-speech detection. In: *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. 86–90.

Gagliardone I, Gal D, Alves T, Martinez G. 2015. Countering online hate speech. *Available at https://unesdoc.unesco.org/ark:/48223/pf0000233231*.

Gao L, Huang R. 2017. Detecting online hate speech using context aware models. In: *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*. Varna, Bulgaria: INCOMA Ltd, 260–266.

Golbeck J, Ashktorab Z, Banjo RO, Berlinger A, Bhagwan S, Buntain C, Cheakalos P, Geller AA, Gnanasekaran RK, Gunasekaran RR, Hoffman KM, Hottle J, Jienjitlert V, Khare S, Lau R, Martindale MJ, Naik S, Nixon HL, Ramachandran P, Rogers KM, Rogers L, Sarin MS, Shahane G, Thanki J, Vengataraman P, Wan Z, Wu DM. 2017. A large labeled corpus

for online harassment research. In: *Proceedings of the 2017 ACM on Web Science Conference*. 229–233.

Hu J, Ruder S, Siddhant A, Neubig G, Firat O, Johnson M. 2020. XTREME: a massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In: *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 4411–4421.

Ibrohim MO, Budi I. 2018. A dataset and preliminaries study for abusive language detection in Indonesian social media. *Procedia Computer Science* **135**:222–229 DOI 10.1016/j.procs.2018.08.169.

Lample G, Conneau A, Denoyer L, Ranzato M. 2018. Unsupervised machine translation using monolingual corpora only. *Available at http://arxiv.org/abs/1711.00043*.

Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **36(4)**:1234–1240.

Leite JA, Silva DF, Bontcheva K, Scarton C. 2020. Toxic language detection in social media for Brazilian Portuguese: new dataset and multilingual analysis. In: *Proceedings of the 2020 Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. Suzhou, China: Association for Computational Linguistics, 914–924.

Lin Y-H, Chen C-Y, Lee J, Li Z, Zhang Y, Xia M, Rijhwani S, He J, Zhang Z, Ma X, Anastasopoulos A, Littell P, Neubig G. 2019. Choosing transfer languages for cross-lingual learning. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence: Association for Computational Linguistics, 3125–3135.

Liu P, Li W, Zou L. 2019. NULI at SemEval-2019 task 6: transfer learning for offensive language detection using bidirectional transformers. In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. 87–91.

Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. 2019. RoBERTa: a robustly optimized BERT pretraining approach. *Available at http://arxiv.org/abs/ 1907.11692*.

Ljubešić N, Erjavec T, Fišer D. 2018. Datasets of Slovene and Croatian moderated news comments. In: *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*. 124–131.

Ljubešić N, Fišer D, Erjavec T. 2019. The FRENK datasets of socially unacceptable discourse in Slovene and English. In: *International Conference on Text, Speech, and Dialogue*. Springer, 103–114.

Lomas N. 2015. Facebook, Google, Twitter commit to hate speech action in Germany. *Available at https://techcrunch.com/2015/12/16/germany-fights-hate-speech-on-social-media/#:~:text=Facebook%2C%20Google%2C%20Twitter%20Commit%20To%20Hate%20Speech%20Action%20In%20Germany,-Natasha%20Lomas%40riptari&text=The%20German%20government%20yesterday%20secured,of%20the%20European%20refugee%20crisis*.

Lomas N. 2017. Facebook, Twitter still failing on hate speech in Germany as new law proposed. TechCrunch. *Available at https://techcrunch.com/2017/03/14/facebook-twitter-still-failing-on-hate-speech-in-germany/*.

Mandl T, Modha S, Majumder P, Patel D, Dave M, Mandlia C, Patel A. 2019. Overview of the HASOC track at FIRE 2019: hate speech and offensive content identification in Indo-European languages. In: *Proceedings of the 11th Forum for Information Retrieval Evaluation*. 14–17.

Martin L, Muller B, Ortiz Suárez PJ, Dupont Y, Romary L, de la Clergerie É, Seddah D, Sagot B. 2020. CamemBERT: a tasty French language model. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics*. 7203–7219.

Mathur P, Shah R, Sawhney R, Mahata D. 2018. Detecting offensive tweets in Hindi–English code-switched language. In: *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*. 18–26.

Miok K, Skrlj B, Zaharie D, Robnik-Sikonja M. 2021. To BAN or not to BAN: Bayesian attention networks for reliable hate speech detection. *Cognitive Computation* DOI 10.1007/s12559-021-09826-9.

Morgan NA. 2020. Update on online harms: written statement—HLWS107. *Available at https://www.parliament.uk/business/publications/written-questions-answers-statements/written-statement/Lords/2020-02-12/HLWS107/*.

Mubarak H, Darwish K, Magdy W. 2017. Abusive language detection on Arabic social media. In: *Proceedings of the First Workshop on Abusive Language Online*. 52–56.

Obadimu A, Mead E, Hussain MN, Agarwal N. 2019. Identifying toxicity within YouTube video comment. In: *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 214–223.

Pamungkas EW, Patti V. 2019. Cross-domain and cross-lingual abusive language detection: a hybrid approach with deep learning and a multilingual lexicon. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*. 363–370.

Pedersen T. 2020. Duluth at SemEval-2020 Task 12: offensive tweet identification in English with logistic regression. In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. Barcelona: International Committee for Computational Linguistics, 1938–1946.

Pelicon A, Pranjić M, Miljković D, Škrlj B, Pollak S. 2020. Zero-shot learning for cross-lingual news sentiment classification. *Applied Sciences* **10(17)**:5993 DOI 10.3390/app10175993.

Pelicon A, Shekhar R, Martinc M, Škrlj B, Purver M, Pollak S. 2021. Zero-shot cross-lingual content filtering: offensive language and hate speech detection. In: *Proceedings of the EACL workshop on News Media Content Analysis and Automated Report Generation*. 30–34.

Pires T, Schlinger E, Garrette D. 2019. How multilingual is multilingual BERT? In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence: Association for Computational Linguistics, 4996–5001.

Plaza-Del-Arco F-M, Molina-González MD, Ureña-López LA, Martn-Valdivia MT. 2020. Detecting misogyny and xenophobia in Spanish tweets using language technologies. *ACM Transactions on Internet Technology (TOIT)* **20(2)**:1–19 DOI 10.1145/3369869.

Poletto F, Basile V, Sanguinetti M, Bosco C, Patti V. 2020. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation* **55(2)**:1–47 DOI 10.1007/s10579-020-09502-8.

Pollak S, Robnik Šikonja M, Purver M, Boggia M, Shekhar R, Pranjić M, Salmela S, Krustok I, Paju T, Linden C-G, Leppänen L, Zosa E, Ulčar M, Freienthal L, Traat S, Cabrera-Diego LA, Martinc M, Lavrač N, Škrlj B, Žnidaršič M, Pelicon A, Koloski B, Podpečan V, Kranjc J, Sheehan S, Boros E, Moreno J, Doucet A, Toivonen H. 2021. EMBEDDIA tools, datasets and challenges: resources and hackathon contributions. In: *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*. Association for Computational Linguistics, 99–109.

Pruksachatkun Y, Phang J, Liu H, Htut PM, Zhang X, Pang RY, Vania C, Kann K, Bowman SR. 2020. Intermediate-task transfer learning with pretrained models for natural language understanding: when and why does it work? In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL*. 5231–5247.

Qian J, Bethke A, Liu Y, Belding E, Wang WY. 2019. A benchmark dataset for learning to intervene in online hate speech. 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)*. Hong Kong: Association for Computational Linguistics, 4755–4764.

Rajpurkar P, Zhang J, Lopyrev K, Liang P. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2383–2392.

Robnik-Sikonja M, Reba K, Mozetic I. 2020. Cross-lingual transfer of twitter sentiment models using a common vector space. In: *Proceedings of the Conference on Language Technologies & Digital Humanities*. Ljubljana: Institute of Contemporary History, 87–92.

Ruder S. 2019. Neural transfer learning for natural language processing. PhD thesis, National University of Ireland, Galway.

Salminen J, Almerekhi H, Milenkovic M, Jung S-g, An J, Kwak H, Jansen BJ. 2018. Anatomy of online hate: developing a taxonomy and machine learning models for identifying and classifying hate in online news media. In: *Proceedings of the Twelfth International AAAI Conference on Web and Social Media (ICWSM 2018)*. 330–339.

Salminen J, Hopf M, Chowdhury SA, Jung S-g, Almerekhi H, Jansen BJ. 2020. Developing an online hate classifier for multiple social media platforms. *Human-centric Computing and Information Sciences* **10(1)**:1 DOI 10.1186/s13673-019-0205-6.

Schmidt A, Wiegand M. 2017. A survey on hate speech detection using natural language processing. In: *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*. 1–10.

Schneider JM, Roller R, Bourgonje P, Hegele S, Rehm G. 2018. Towards the automatic classification of offensive language and related phenomena in German tweets. In: *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*. 95–103.

Schuster M, Nakajima K. 2012. Japanese and Korean voice search. In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Piscataway: IEEE, 5149–5152.

Shekhar R, Pranjić M, Pollak S, Pelicon A, Purver M. 2020. Automating news comment moderation with limited resources: benchmarking in Croatian and Estonian. *Journal for Language Technology and Computational Linguistics (JLCL)* **34(1)**:49–79.

Simonite T. 2020. Facebook's AI for hate speech improves. how much is unclear. *WIRED*. *Available at https://www.wired.com/story/facebook-ai-hate-speech-improves-unclear/*.

Škrlj B, Eržen N, Sheehan S, Luz S, Robnik-Šikonja M, Pollak S. 2021. *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*. Association for Computational Linguistics, 76–83.

Stappen L, Brunn F, Schuller B. 2020. Cross-lingual zero-and few-shot hate speech detection utilising frozen transformer language models and AXEL. *Available at http://arxiv.org/abs/2004.13850*.

Stevenson A. 2018. Facebook admits it was used to incite violence in Myanmar. *New York Times*. *Available at https://www.nytimes.com/2018/11/06/technology/myanmar-facebook.html*.

Struß JM, Siegel M, Ruppenhofer J, Wiegand M, Klenner M. 2019. Overview of GermEval task 2, 2019 shared task on the identification of offensive language. In: *German Society for Computational Linguistics. Proceedings of the 15th Conference on Natural Language Processing (KONVENS) 2019*. Nürnberg/Erlangen: s.a., 354–365.

Subedar A. 2018. The country where Facebook posts whipped up hate. In: *Available at https://www.bbc.co.uk/news/blogs-trending-45449938*.

Swayamdipta S, Schwartz R, Lourie N, Wang Y, Hajishirzi H, Smith NA, Choi Y. 2020. Dataset cartography: mapping and diagnosing datasets with training dynamics. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Ulčar M, Robnik-Šikonja M. 2020. FinEst BERT and CroSloEngual BERT. In: *International Conference on Text, Speech, and Dialogue*. Springer, 104–111.

van der Goot R, Ljubešić N, Matroos I, Nissim M, Plank B. 2018. Bleaching text: abstract features for cross-lingual gender prediction. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*. 383–389.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. 2017. Attention is all you need. In: *Advances in neural information processing systems*. 5998–6008.

Vidgen B, Botelho A, Broniatowski D, Guest E, Hall M, Margetts H, Tromble R, Waseem Z, Hale S. 2020. Detecting east Asian prejudice on social media. In: *Proceedings of the Fourth Workshop on Online Abuse and Harms*. Association for Computational Linguistics, 162–172.

Vidgen B, Derczynski L. 2020. Directions in abusive language training data, a systematic review: garbage in, garbage out. *PLOS ONE* **15(12)**:e0243300 DOI 10.1371/journal.pone.0243300.

Vu T, Wang T, Munkhdalai T, Sordoni A, Trischler A, Mattarella-Micke A, Maji S, Iyyer M. 2020. Exploring and predicting transferability across NLP tasks. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 7882–7926.

Wang A, Hula J, Xia P, Pappagari R, McCoy RT, Patel R, Kim N, Tenney I, Huang Y, Yu K, Jin S, Chen B, Van Durme B, Grave E, Pavlick E, Bowman SR. 2019a. Can you tell me how to get past Sesame Street? Sentence-level pretraining beyond language modeling. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence: Association for Computational Linguistics, 4465–4476.

Wang A, Pruksachatkun Y, Nangia N, Singh A, Michael J, Hill F, Levy O, Bowman S. 2019b. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In: *Advances in Neural Information Processing Systems*. 3266–3280.

Wang A, Singh A, Michael J, Hill F, Levy O, Bowman SR. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels, Belgium: Association for Computational Linguistics, 353–355.

Wiegand M, Siegel M, Ruppenhofer J. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. In: *Proceedings of the GermEval 2018 Workshop (GermEval)*.

Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M, Davison J, Shleifer S, von Platen P, Ma C, Jernite Y, Plu J, Xu C, Scao TL, Gugger S, Drame M, Lhoest Q, Rush AM. 2020. HuggingFace's transformers: state-of-the-art natural language processing. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 38–45.

Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K, Klingner J, Shah A, Johnson M, Liu X, Kaiser u, Gouws S, Kato Y, Kudo T, Kazawa H, Stevens K, Kurian G, Patil N, Wang W, Young C, Smith J, Riesa J, Rudnick A, Vinyals O, Corrado G, Hughes M, Dean J. 2016. *Available at http://arxiv.org/abs/1609.08144*.

Wulczyn E, Thain N, Dixon L. 2017. Ex Machina: personal attacks seen at scale. In: *Proceedings of the 26th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee*. 1391–1399.

Yogatama D, d'Autume CdM, Connor J, Kocisky T, Chrzanowski M, Kong L, Lazaridou A, Ling W, Yu L, Dyer C, Blunsom P. 2019. Learning and evaluating general linguistic intelligence. arXiv preprint. *Available at http://arxiv.org/abs/1901.11373*.

Zampieri M, Malmasi S, Nakov P, Rosenthal S, Farra N, Kumar R. 2019a. Predicting the type and target of offensive posts in social media. In: *Proceedings of NAACL*. 1415–1420.

Zampieri M, Malmasi S, Nakov P, Rosenthal S, Farra N, Kumar R. 2019b. SemEval-2019 task 6: identifying and categorizing offensive language in social media (OffensEval). *Proceedings of the 13th International Workshop on Semantic Evaluation*. Minneapolis, Minnesota, USA: Association for Computational Linguistics, 75–86.

Zampieri M, Nakov P, Rosenthal S, Atanasova P, Karadzhov G, Mubarak H, Derczynski L, Pitenis Z, Çöltekin C. 2020a. SemEval-2020 Task 12: multilingual offensive language identification in social media (OffensEval 2020). In: *Proceedings of SemEval.*

Zampieri M, Nakov P, Rosenthal S, Atanasova P, Karadzhov G, Mubarak H, Derczynski L, Pitenis Z, Çöltekin Ç. 2020b. SemEval-2020 task 12: multilingual offensive language identification in social media (OffensEval 2020). In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. Barcelona: International Committee for Computational Linguistics, 1425–1447.

Appendix C: Not All Comments are Equal: Insights into Comment Moderation from a Topic-Aware Model

Not All Comments are Equal: Insights into Comment Moderation from a Topic-Aware Model

Elaine Zosa
University of Helsinki
elaine.zosa@helsinki.fi

Ravi Shekhar
Queen Mary University of London
r.shekhar@qmul.ac.uk

Mladen Karan
♢Queen Mary University of London
m.karan@qmul.ac.uk

Matthew Purver♢,†
†Jožef Stefan Institute
m.purver@qmul.ac.uk

Abstract

Moderation of reader comments is a significant problem for online news platforms. Here, we experiment with models for automatic moderation, using a dataset of comments from a popular Croatian newspaper. Our analysis shows that while comments that violate the moderation rules mostly share common linguistic and thematic features, their content varies across the different sections of the newspaper. We therefore make our models topic-aware, incorporating semantic features from a topic model into the classification decision. Our results show that topic information improves the performance of the model, increases its confidence in correct outputs, and helps us understand the model's outputs.

1 Introduction

Most newspapers publish their articles online, and allow readers to comment on those articles. This can increase user engagement and page views, and provides readers with an important route to public freedom of expression and opinion, with the ability to interact and discuss with others. Comment sections usually provide some degree of anonymity;[1] while improving accessibility, this can also encourage inappropriate behaviour, and publishers therefore usually employ some moderation policy to regulate content and to ensure legal compliance (in some cases, publishers can be held responsible for user-contributed content on their sites).

One possible approach is a 'moderate then publish' policy, in which comments must be approved by a moderator before they appear; this requires significant manpower and introduces delays and limitations into the user conversation (for example, the New York Times only allows comments for one day after article publication[2]). On the other hand, a 'publish then moderate' strategy, in which comments are published immediately, and later removed if necessary, is less effective at blocking toxic or illegal content. Combined with the increase in comment volumes in recent years there is increasing interest in automatic moderation methods (see e.g. Pavlopoulos et al., 2017a), either as standalone tools or for integration into human moderators' practices (Schabus and Skowron, 2018).

Detecting comments that need moderators' attention is usually approached as a text classification task (see e.g. Pavlopoulos et al., 2017a); but comments can be blocked for a range of reasons (Shekhar et al., 2020). One is the presence of offensive language, a well-studied NLP task (see Section 2 below); however, others include advertising or spam, illegal content, spreading misinformation, trolling and incitement — all distinct categories which might be expected to show distinct features, and perhaps to vary according to the content being commented on. Another aspect that distinguishes the comment moderation task from the usual text classification tasks in NLP is the need for interpretable or explainable models: if classifiers are to be used by human moderators within publishers' working practices, they must be able to understand the outputs (Švec et al., 2018).

Here, we therefore investigate models which can provide both an aspect of interpretability and the ability to take account of the topics being discussed, by incorporating topic information into the comment classifier. Specifically, we incorporate semantic representations learned by the Embedded Topic Model (ETM) (Dieng et al., 2020) into a classifier pipeline based on Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997). Our model improves performance

[1] Some newspapers allow completely anonymous posting; some require commenters to create an account with a username, but this does not usually reveal their true identity.

[2] NYT Comment FAQ: https://nyti.ms/2PF02kj

Proceedings of Recent Advances in Natural Language Processing, pages 1652–1662
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_185

by 4.4% over a text-only approach on the same dataset (Shekhar et al., 2020), and is more confident in the correct decisions it makes. Inspection of the topic distributions reveals how different newspaper sections have different language and topic distributions, including differences in the kind of comments that need moderation.[3]

2 Related Work

Automated news comment moderation Most research on this task so far formulates it as a text classification problem: for a given comment, the model must predict whether the comment violates the newspaper's policy. However, approaches to classification vary. Nobata et al. (2016) use a range of linguistic features, e.g. lexicon and n-grams. Pavlopoulos et al. (2017a) and Švec et al. (2018) use neural networks, specifically RNNs with an attention mechanism. Recently, Tan et al. (2020) and Tran et al. (2020) apply a modified BERT model (Devlin et al., 2019) while Schabus et al. (2017) use a bag-of-words approach.

Some approaches go beyond the comment text itself: Gao and Huang (2017) add information like user ID and article headline into their RNN to make the model context-aware; Pavlopoulos et al. (2017b) incorporate user embeddings; Schabus and Skowron (2018) incorporate the news category metadata of the article. However, no work so far investigates automatic modelling of topics (rather than relying on categorical metadata), or applies this to the comments rather than just their parent articles.

Some steps towards model intepretability and output explanation have also been taken: both Švec et al. (2018) and Pavlopoulos et al. (2017a) use an attention saliency map to highlight possibly problematic words. However, we are not aware of any work using higher-level topic information as a route to understanding model outputs.

Available datasets Several datasets have been created for the news comment moderation task. Nobata et al. (2016) provide 1.43M comments posted on Yahoo! Finance and News over 1.5 years, in which 7% of the comments are labelled as abusive via a community moderation process. Gao and Huang (2017) contains 1.5k comments from Fox News, annotated with specific hateful/non-hateful labels as a post-hoc task, and having 28% hateful comments. However, both are relatively small, and their labelling methods mean that neither dataset is entirely representative of the moderation process performed by newspapers.

Pavlopoulos et al. (2017a) provides 1.6M comments from Gazzetta, a Greek sports news portal, over c.1.5 years. Here, 34% of comments are labelled as blocked, and the labels are derived from the newspaper's human moderators and journalists. Schabus et al. (2017) and Schabus and Skowron (2018) provide a dataset from a German-language Austrian newspaper with 1M comments posted over 1 year, out of which 11,773 comments are annotated using seven different rules.

More recently, Shekhar et al. (2020) present a dataset from 24sata, Croatia's most widely read newspaper.[4] This dataset is significantly larger (10 years, c.20M comments); and moderator labels include not only a label for blocked comments, but also a record of the reason for the decision according to a 9-class moderation policy. However, their experiments show that classifier performance is limited, and transfers poorly across years. Here, we therefore use this dataset (see Section 3), with a view to improving performance and applying a topic-aware model to improve and better understand the robustness in the face of changing topics.

Related tasks More attention has been given to related tasks, most prominently the detection of offensive language, hate speech, and toxicity (Pelicon et al., 2021). A comprehensive survey of dataset collection is provided by Poletto et al. (2020) and Vidgen and Derczynski (2020).[5]

Topic Modelling Topic models capture the latent themes (also known as *topics*) from a collection of documents through the co-occurence statistics of the words used in a document. Latent Dirichlet Allocation (LDA) (Blei et al., 2003), a popular method for capturing these topics, is a generative document model where a document is a mixture of topics expressed as a probability distribution over the topics and a topic is a distribution over the words in a vocabulary. The Embedded Topic Model (ETM, Dieng et al., 2020) is an LDA-like topic modelling method that exploits the semantic information captured in word embeddings during topic inference. The advantage of ETM over LDA

[3] Source code available at https://github.com/ezosa/topic-aware-moderation

[4] http://24sata.hr/

[5] http://hatespeechdata.com/ provides a comprehensive list of relevant datasets.

Comment Moderation Data			
	Blocked	Non-blocked	Blocking Rate
Train	4984	75016	6.23%
Valid	642	9358	6.42%
Test	37271	438142	7.84%
Topic Modelling Data			
	Blocked	Non-blocked	Blocking Rate
Train	34863	36725	48.70%
Valid	4880	5120	48.80%

Table 1: Details of datasets used in experiments.

Section (– Subsection)	Blocked	Non-blocked	Blocking Rate
Kolumne (*Columns*)	655	6382	9.31%
Lifestyle	2426	30985	7.26%
Show	6827	58896	10.39%
Sport	5882	80820	6.78%
Tech	382	7173	5.06%
Vijesti (*News*)	20094	239835	7.73%
– Crna kronika (*Crime*)	5917	62471	8.65%
– Hrvatska (*Croatia*)	3527	45170	7.70%
– Politika (*Politics*)	6088	80264	7.05%
– Svijet (*World*)	2625	31459	7.24%

Table 2: Details per section, and (for section Vijesti) sub-section, of the comment moderation test set.

is that it combines the advantages of word embeddings with the document-level dependencies captured by topic modelling and has been shown to produce more coherent topics than regular LDA.

3 Dataset

We use the 24sata comment dataset (Shekhar et al., 2020; Pollak et al., 2021), introduced in Section 2. This contains c.21M comments on 476K articles from the years 2007-2019[6], written in Croatian. The dataset has details of comments blocked by the 24sata moderators, based on a set of moderation rules–these vary from hate speech to abuse to spam (see Shekhar et al., 2020, for rule description). The dataset also identifies the article under which a comment was posted, together with the section/subsection of the newspaper the article appeared in. These sections/sub-sections relate to the content of the article: for example, the Sport section contains sports-related news while the Kolumne (*Columns*) section contains opinion pieces. The largest section, Vijesti (*News*), is further subdivided as shown in Table 2.

3.1 Data Selection

In this work, we use data from 2018 for training and validation of the topic model and classifiers and data from 2019 for testing. This reflects the realistic scenario where we use data collected from the past to make predictions. For training and validation, we randomly select 50,000 articles out of 65,989 articles from 2018, sampling from the nine most-representative sections/sub-sections (Table 2). Each article comes with c.50 comments on average.

To train the topic model, we sample around 80,000 comments across these articles, with a roughly equal split between blocked and non-blocked comments. This is to encourage a diverse mix of topics from both comment classes. As a preprocessing step we remove comments with less than 10 words from the training data (see Table 1 (lower part)). To train the classifiers, we randomly sample around 80,000 comments such that the sampled set has the same blocking rate as the entire 2018 dataset.

For the test set, we then use all 475,413 comments associated with the 17,953 articles from 2019. Table 1 (upper part) provides the dataset details, with comment moderation blocking rate. For the test set, Table 2 provides details on the section and sub-section of the related articles. These top nine sections account for more than 95% of the comments of the entire test set.

3.2 Content Analysis

To gain some insight into the content of blocked comments, we analyze the linguistic differences between blocked and non-blocked comments and across different sections. First, we compare comment length. As we can see from Table 3, blocked and non-blocked comments have, on average, similar lengths. However, if we further divide blocked comments into two sub-groups — spam and non-spam — we find that on average, spam comments are longer than other comments. We observe a similar pattern across different sections.

Next, we measure lexical diversity using mean-segmental type-token ratio (MSTTR). The MSTTR is computed as the mean of type-token ratio for every 1000 tokens in a dataset to control for dataset size (van Miltenburg et al., 2018). From Table 3, we see that non-blocked comments have higher MSTTR (i.e. higher lexical diversity) than blocked comments (0.62 vs 0.46). However, when we again divide blocked comments into spam and non-spam,

[6]Dataset is available at http://hdl.handle.net/11356/1399

we observe that non-spam blocked comments have a similar MSTTR to non-blocked comments (0.61 vs 0.62), while spam comments have much lower MSTTR (0.35 vs 0.61). This suggests that blocked comments (excluding spam) have as rich a vocabulary as non-blocked. Again, we see a similar pattern across different news sections.

	Mean length	MSTTR
All	23.06	0.61
Non-blocked	23.01	0.62
Blocked	23.65	0.46
Blocked (non-spam)	19.16	0.61
Blocked (Spam only)	28.23	0.35

Table 3: Mean-segmental TTR and average length of comments

Now we look at the top bigrams of each class. We collect all bigrams that occur at least 50 times and rank them according to their pointwise mutual information (PMI) score. In general, we do not see many overlaps between the top bigrams of blocked and non-blocked comments across the different sections. Bigrams in blocked comments indicate spam messages such 'iskustva potrebnog' (*experience required*), 'redoviti student' (*full-time student*) and 'prilika pružila' (*opportunity given*). Removing spam comments, we encounter bigrams used for swearing such as 'pas mater' (*damn it*) and 'jedi govna' (*eat sh*t*). In the non-blocked comments, the top bigrams are more relevant to the section they appear in. For instance, in the Vijesti section, top bigrams include 'new york', 'porezni obveznici' (*taxpayers*) and 'naftna polja' (*oil fields*) while in Sports, top bigrams include 'all star', 'grand slam' and 'man utd'.

This suggests that the content of blocked comments tends to share commonalities across sections more than non-blocked comments; but again, these commonalities may be mostly within the spam category, with other blocked categories being more topic-dependent. Our next step therefore is to examine the use of topic modelling to capture these dependencies, with a view to using topic information to improve a moderation classifier.

4 Topic Modelling

We now apply a topic model to gain insight into what characterises a blocked comment and a non-blocked one, and whether this varies between different sections where different subjects are discussed.

4.1 Topic Model

We use the Embedded Topic Model (ETM, Dieng et al., 2020) as our topic model since it has been shown to outperform regular LDA and and other neural topic modelling methods such as NVDM (Miao et al., 2016). We also want to take advantage of ETM's ability to incorporate the information encoded in pretrained word embeddings trained on vast amounts of data to produce more coherent topics. In the ETM, the topic-term distribution for topic k, β_k, is induced by a matrix of word embeddings ρ and its respective topic embedding α_k which is a point in the word embedding space:

$$\beta_k = softmax(\rho^T \alpha_k) \quad (1)$$

The topic embeddings are learned during topic inference while the word embeddings can be pretrained or also learned during topic inference. In this work, we use pretrained embeddings.

The document-topic distribution of a document d, θ_d, is drawn from the logistic normal distribution whose mean and variance come from an inference network:

$$\theta_d \sim LN(\mu_d, \sigma_d) \quad (2)$$

Given a trained ETM, we can infer the **document-topic distribution (DTD)** of an unseen document. In addition, we can also compute a **document-topic embedding (DTE)** as the weighted sum of the embeddings of the topics in a document, where the weight corresponds to the probability of the topic in that document:

$$DTE = \sum_{k=0}^{K} \alpha_k \theta_{d,k} \quad (3)$$

where α_k is the topic embedding of topic k, and $\theta_{d,k}$ is the probability of topic k in doc d.

4.2 Topic Analysis

Now we analyse the usage of topics in our test set. We trained the ETM for 100 topics on the training set and inferred the topic distributions of the comments in the test set. For analysis, we extract the top topics in a set of comments. To do this, we take the mean of the topic distributions over the comments in the set and rank the topics according to their weight in this mean distribution. We then take the top 15 topics for analysis because this is the average number of topics in a comment with a non-zero probability in our test set. Note that in this analysis we only use the document-topic

distributions and not the document-topic embeddings. To more easily discuss the topics here we provide concise labels for each topic as interpreted by a native speaker. Automatic labelling of topics is a non-trivial task and an area of active research (Bhatia et al., 2016; Alokaili et al., 2020; Popa and Rebedea, 2021).

First, we examine the prevalent topics in the blocked and non-blocked comments, separately. The top topics of non-blocked comments cover a diverse range of subjects from politics to football while the top topics in blocked comments are dominated by spam and offensive language (Figure 1). However, we also see many topics shared between blocked and non-blocked comments. [7].

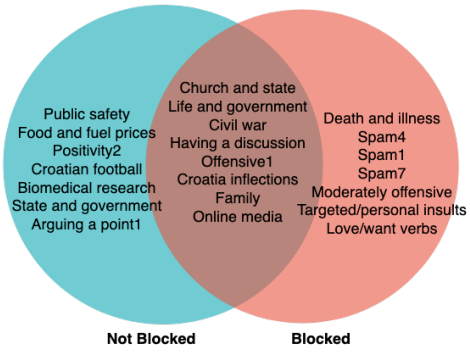


Figure 1: Top topics of the blocked and non-blocked comments for the entire test set.

Next we illustrate how different topics intersect and diverge between blocked and non-blocked comments across sections by looking at the top topics of two thematically-different sections, Lifestyle and Politika (*Politics*).

Figure 2 shows the top topics of these sections and the intersections between them. In Politics, blocked comments tend toward spam and targeted insults. Non-blocked topics include public safety and finances. However, we also see that more than half of the top topics overlap between blocked and non-blocked. This suggests that, thematically, there isn't a very clear distinction between blocked and non-blocked comments in the Politics section.

In Lifestyle, blocked topics are dominated by spam and while there are topics on offensive insults, they are not as prevalent as the spam-related ones. The non-blocked topics are about family and relationships and commenters arguing with each other. Compared to Politics, we see a clearer distinction between topics in blocked and non-blocked in this section. In terms of topic overlaps between Lifestyle and Politics, blocked comments in both sections are dedicated to spam and insults while non-blocked comments focus on positive sentiments.

The combination of certain topics also provide an indication of the classification of the comment. For instance, we notice the use of topics about football cards in comments that do not do not discuss the sport (for instance, football cards as a topic is prominent in the blocked Lifestyle comments). It turns out that some commenters use the red and yellow cards from football as metaphors for being banned or having their comments blocked by moderators (12% of comments that use these metaphors are blocked by moderators). On the other hand, comments that use the football cards topics *and* any of the sports-related topics are likely to be a genuine discussion of football (only 5% of such comments are blocked by moderators). We show some examples of these comments in Table 5.

So clearly there is a distinction between the usage of topics in the non-blocked and blocked comments. We therefore think it is a good idea to propose a model which incorporates topic information into a comment moderation classifier.

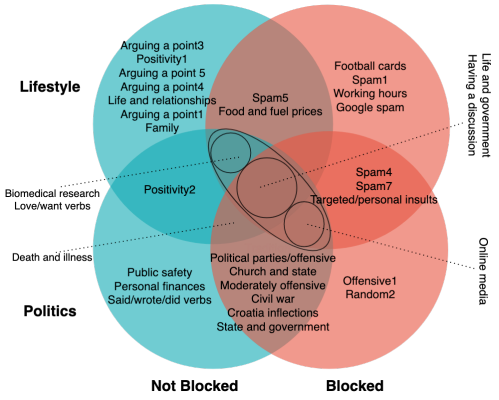


Figure 2: Top topics of the blocked and non-blocked comments in the Lifestyle and Politics sections.

5 Topic-aware Classifier

Our aim is to improve comment moderation predictions by combining textual features with document-level semantic information in the form of topics. To this end, we test several model architectures that combine a language model with topic features.

For the comment text representation, we use a

[7] All 100 topics and labels are available at https://github.com/ezosa/topic-aware-moderation

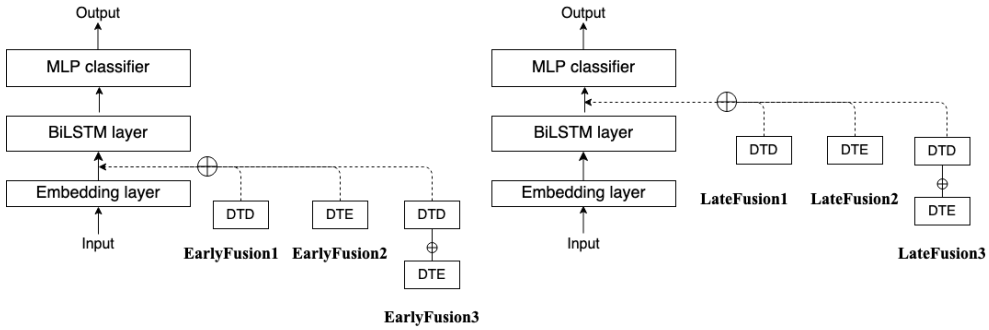


Figure 3: Architectures combining text and topic features. DTD is the topic distribution of a document while DTE is the topic embedding.

bidirectional LSTM (BiLSTM, Schuster and Paliwal, 1997). The comment text is given as input to an embedding layer then a BiLSTM layer where the output of the final hidden state is taken as the encoded representation of the comment. For the topic representations, we use the topic distributions (DTD) and topic embeddings (DTE) discussed in Section 4.1.

We propose two fusion mechanisms to combine the text and topic representations: *early* and *late* fusion. In early fusion, topic features are concatenated with the output of the embedding layer and then passed to the BiLSTM layer. In **EarlyFusion1 (EF1)**, only DTD is concatenated with the word embeddings; **EarlyFusion2 (EF2)** uses DTE instead of DTD; and **EarlyFusion3 (EF3)** uses both DTE and DTD. In late fusion, topic features are concatenated with the output representation of the BiLSTM layer, and passed to the MLP for classification. Again, **LateFusion1 (LF1)** uses DTD; **LateFusion2 (LF2)** uses DTE; and **LateFusion3 (LF3)** uses both. Figure 3 shows the architectures.

Our model is inspired by the Topic Compositional Neural Language Model (TCNLM, Wang et al., 2018) and the Neural Composite Language Model (NCLM, Chaudhary et al., 2020) that incorporate latent document-topic distributions with language models. Both of these models simultaneously learn a topic model and a language model through a joint training approach. The NCLM introduced the use of word embeddings to generate an explanatory topic representation for a document in addition to the document-topic proportions. In our work, instead of using the word embeddings of the top words of the latent topics of a document (where the number of top words is a hyperparameter), we leverage the topic embeddings learned by ETM and combine them with the document-topic proportions to produce the document-topic embeddings (DTE). Also unlike the TCNLM and NCLM, we use pre-trained topics in our model so as to easily de-couple and analyse the influence of topics in the classifier performance. Another related work is TopicRNN (Dieng et al., 2016), a model that uses topic proportions to re-score the words generated by the language model. The topics generated by this model, however, have been shown to have lower coherences compared to NCLM (Chaudhary et al., 2020).

6 Experimental Setup

Dataset As discussed in Section 3.1, we use the 2018 data as the training and validation sets of our topic-aware classifier and the 2019 data as the test set. Details of the train and validation sets are shown in Table 1 and the test set in Table 2.

Baseline models To assess how topic information improves comment classification, we use as baselines the following models trained only on text *or* topics:

- **Text only**: a classifier with BiLSTM & MLP layers, similar to Figure 3 but with comment text alone as input.
- **Document-topic distribution (DTD)**: MLP only, document-topic distributions as input.
- **Document-topic embedding (DTE)**: MLP only, document-topic embeddings as input.
- **DTD+E**: MLP only, concatenated document-topic distributions and embeddings.

Hyperparameters We use 300D word2vec embeddings, pretrained on the Croatian Web Corpus (HrWAC, Ljubešić and Erjavec, 2011; Šnajder, 2014), for training the ETM and to initialize the embedding layer of the BiLSTM. The ETM is trained

for 500 epochs for 100 topics using the default hyperparameters from the original implementation [8]. The BiLSTM is composed of one hidden layer of size 128 with dropout set to 0.5. The MLP classifier is composed of one fully-connected layer, one hidden layer of size 64, a ReLU activation, and a sigmoid for classification with the classification threshold set to 0.5. We use Adam optimizer with $lr = 0.005$. We train all classifiers for 20 epochs with early stopping based on the validation loss.

7 Results

In Table 4, we present the performance of the baselines and proposed models, measured as macro F1-scores. All models that combine text and topic representations perform better than the models that use only text *or* topics. Of the baseline models, the DTD model performs comparatively better than the DTE and DTD+E models, and surprisingly performs almost as well as the Text-only model; however, we show in Section 8 below that DTD is much less confident in its predictions than the Text-only model. Overall, the best performing model is LF1, which improves the Text-only model's performance by $+4.4\%$ (67.37% vs 62.97%); and improves by a similar amount over Shekhar et al.'s results using mBERT (macro-F1 score 62.07 for year 2019).

Interestingly, we see a wide variation in performance across news sections. We observe that comments in Lifestyle and Tech are the easiest to classify (best F1 over 72.00) while Politika (*Politics*) is the most difficult (best F1 around 61.61). The main cause appears to be that Lifestyle and Tech have the highest proportion of spam comments: on average, 49.44% of blocked comments in the test set are spam, but for Lifestyle and Tech this number rises to 77.25% and 69.63%, respectively. As for the Politics section, the most likely reason the comments are difficult to classify is that, excluding spam, there is a high degree of overlap in the subjects discussed in the blocked and non-blocked comments (see the topic analysis in Section 4.2).

7.1 Analysis of Classifier Outputs

In general, we observe that blocked comments tend to use similar topics across different sections while non-blocked comments have more diverse topics. Of the nine sections that we analyzed, there are five topics that are prominent in blocked comments in all sections ('Targeted/personal insults', 'Spam4', 'Spam7', 'Online media', and, 'Having a discussion') and only three topics prominent in non-blocked comments ('Having a discussion', 'Online media', and, 'Life and government'). This suggests that blocked comments are more semantically-coherent across sections than non-blocked ones. In contrast, topics in non-blocked comments tend to be more relevant to their respective sections: for instance, family and relationships are not discussed a lot in the Politics section, while Lifestyle commenters do not tend to talk about political issues.

The higher topical coherence then of blocked comments explains why a text classification approach can achieve reasonable performance; but the variation in blocked comment content between some sections explains why adding topic information improves our classification results.

Next, we analyze the confidence of classifiers and examine some of the outputs of the models. To analyze confidence, we gradually increase the classification threshold from 0.5 to 1.0 in increments of 0.05. For every new threshold, we plot the macro-F1 for the different models (Figure 4). We compare the confidence of four models: DTD, Text-only, EF2 (the strongest early fusion model), and LF1 (the overall best-performing model). We find that the most confident model is LF1 and the least confident is DTD. The two fusion classifiers display similar levels of confidence. The Text-only classifier is not as confident as the fusion classifiers but still more confident than DTD. This suggests that adding topic features to text not only improves performance, it also increases classifier confidence.

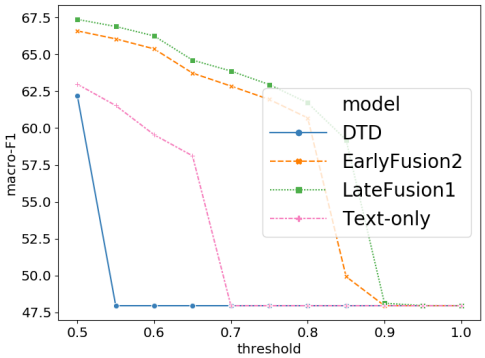


Figure 4: Confidence of the top performing models.

In Table 5 we give some examples of comments and the classifier decisions of the Text-only classifier and LF1 (our best-performing fusion model) and their top topics (topics with $prob > 0.10$). The

[8] https://github.com/adjidieng/ETM

Section – Subsection	Text only	Topics only			Text+Topic Combinations					
		DTD	DTE	DTD+E	EF1	EF2	EF3	LF1	LF2	LF3
All	62.97	62.20	59.3	58.33	66.33	66.58	65.61	**67.37**	66.22	66.95
Kolumne	59.86	59.65	56.25	55.33	62.40	62.90	63.13	63.25	62.38	**63.6**
Lifestyle	69.21	70.07	65.93	64.47	72.73	70.9	69.36	72.00	72.39	**72.92**
Show	61.97	61.30	58.62	57.60	65.24	65.63	64.26	**66.50**	65.00	65.86
Sport	63.22	61.42	58.61	57.90	67.11	67.86	66.74	**68.26**	67.14	67.82
Tech	64.87	66.37	63.17	62.55	67.72	68.74	67.65	68.76	67.68	**69.15**
Vijesti (*News*)	62.38	61.49	58.79	57.77	65.58	65.99	65.24	**66.77**	65.53	66.24
– Crna kronika	64.67	63.98	61.03	59.84	68.10	68.88	68.11	**69.60**	67.89	68.88
– Hrvatska	63.61	63.50	60.10	58.93	67.24	66.86	65.95	67.90	67.12	**67.95**
– Politika	57.93	56.49	54.95	54.20	60.51	61.52	60.84	**61.61**	60.63	61.30
– Svijet	63.58	62.55	59.62	58.35	66.83	66.95	66.33	68.44	67.21	67.57

Table 4: Classifier performance measured as macro-F1.

Comment	Label	Text-only	LF1	Top topics
1. konačno. gamad lopovska crno bijela prevarantska (*finally. the black and white cheating thieving bastards*)	1	1 (0.501)	1 (0.687)	Arguing a point, Political parties (offensive)
2. ...dobro jutro,moze crveni karton za novinara koji je osmislio naslov ;-) (*... good morning, how about a red card for the journalist who came up with this title ;-)*)	1	0 (0.315)	0 (0.456)	Football cards
3. Ne bum komentiral, dosta mi je kazni od žutih i crvenih kartona. Strah me je cenzure i bradate cure. (*No comment, I'm tired of getting yellow and red cards. I'm afraid of censorship and bearded ladies.*)	0	0 (0.054)	0 (0.335)	Football cards, Random
4. Koji kurac Rumunjski sudac ne da koji karton više Čehima. Pa svake tri minute sa leđa sruše Olma !!!! *(Why the fuck does the Romanian referee not give a few cards more to the Czechs, They tackle Olm from behind every three minutes.)*	0	0 (0.303)	1 (0.587)	Targeted/personal insults
5. Baš ste jadnici kao i ovi sa 24sata koji u ovome uživaju ! (*All of you are lame as well as those from 24sata who enjoy this.*)	1	0 (0.171)	0 (0.229)	Online media, Moderately offensive
6. Google sada plaća između 15.000 i 30.000 dolara mjesečno za rad na mreži od kuće. Pridružio sam se ovom poslu prije 3 mjeseca i zaradio 24857 dolara u prvom mjesecu ovog posla. >>> URL (*Google now pays between 15.000 and 30.000 dollars per month for working remotely from home. I started this job 3 months ago and made 24857 dollars in the first month of this job. >>> URL*)	0	1 (0.67)	1 (0.90)	Spam4

Table 5: Sample comments and classifier decisions.

first example contains swearing which both models pick up on and classify as blocked although LF1 is more confident in its decision then Text-only. In the second example, both models predict the wrong label but LF1 treats this as a borderline case because it is targeted at the moderators. However since this is only a mild provocation of the moderators, this might be a case where the gold label is incorrect. The topics also pick up on the fact that this comment talks about football cards but only has a tenuous connection to the sport ("getting a red card" is an expression used for "being banned"). In contrast, the third comment also uses the banning sense of "card" but is not directed at anyone, and is thus labeled as 0 (non-blocked), which both models get right. Again the topics indicate that the comment is not really about the sport. The fourth example shows a case where "cards" are mentioned in their standard football sense but also contains a swear word, making the gold label of 0 (non-blocked) questionable. The better performance of LF1 on such examples, compared to Text-only, implies that

LF1 is better aware of the different semantics of "card" (sports-related vs. metaphorical), likely due to added topic information.

The fifth example contains a moderately offensive insult that is not directed at any single group except the 24sata readership in general. One reason why both classifiers do not get this right is that the word *jadnici* is not strong enough to be considered offensive. Finally the last example is clearly a spam comment that both classifiers correctly classify but for which the gold label is incorrect.

Overall, compared to the Text-only model, we find that LF1 more often than not improves the confidences (and sometimes the classification), especially in cases in which the gold label is clear. This is valuable in practice, as better confidences might lead to better prioritisation of comments for manual moderation, reducing the time required to remove the most problematic ones.

8 Conclusion

In this work, we propose a model to combine document-level semantics in the form of topics with text for comment moderation. Our analysis shows that blocked and non-blocked comments have different linguistic and thematic features, and that topics and language use vary considerably across news sections, including some variation in the comments that should be blocked. We also found that blocked comments tend to be more semantically coherent across sections than non-blocked ones. We therefore see that the use of topics in our model improves performance, and gives more confident outputs, over a model that only uses the comment text. The model also provides topic distributions, interpretable as keywords, as a form of an explanation of its prediction. As future work, we plan to incorporate comment, article, and user metadata into the model.

Acknowledgements

This work has been supported by the European Union Horizon 2020 research and innovation programme under grants 770299 (NewsEye) and 825153 (EMBEDDIA), and by the UK EPSRC under grant EP/S033564/1.

Ethics and Impact Statement

Data The dataset and annotations are provided by the publisher of 24sata.hr, Styria Media Group, for research purposes and deposited in the CLARIN repository. The authors of the comments are anonymised. The researchers used the data as-is and did not modify or add annotations.

Intended Use The models we present here are intended to assist comment moderators in their work. We do not recommend that the model be deployed in the moderation process without a human-in-the-loop.

Potential Misuse The models and the analysis of their performance we provide in this paper could be used by malicious actors to gain an insight into the comment moderation process and find loopholes in the process. However, we think such a risk is unlikely and the impact it might have outweighs the potential benefits of models intended to assist human moderators such as the ones we present here.

References

Areej Alokaili, Nikolaos Aletras, and Mark Stevenson. 2020. Automatic generation of topic labels. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1965–1968.

Shraey Bhatia, Jey Han Lau, and Timothy Baldwin. 2016. Automatic labelling of topics with neural embeddings. *arXiv preprint arXiv:1612.05340*.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

Yatin Chaudhary, Hinrich Schütze, and Pankaj Gupta. 2020. Explainable and discourse topic-aware neural language understanding. In *International Conference on Machine Learning*, pages 1479–1488. PMLR.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Adji B Dieng, Chong Wang, Jianfeng Gao, and John Paisley. 2016. Topicrnn: A recurrent neural network with long-range semantic dependency. *arXiv preprint arXiv:1611.01702*.

Lei Gao and Ruihong Huang. 2017. Detecting online hate speech using context aware models. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 260–266, Varna, Bulgaria. INCOMA Ltd.

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

Nikola Ljubešić and Tomaž Erjavec. 2011. hrWaC and slWaC: Compiling web corpora for Croatian and Slovene. In *International Conference on Text, Speech and Dialogue*, pages 395–402. Springer.

Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *International conference on machine learning*, pages 1727–1736. PMLR.

Emiel van Miltenburg, Ruud Koolen, and Emiel Krahmer. 2018. Varying image description tasks: spoken versus written descriptions. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 88–100.

Chikashi Nobata, J. Tetreault, A. Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. *Proceedings of the 25th International Conference on World Wide Web*.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017a. Deeper attention to abusive user content moderation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1125–1135, Copenhagen, Denmark. Association for Computational Linguistics.

John Pavlopoulos, Prodromos Malakasiotis, Juli Bakagianni, and Ion Androutsopoulos. 2017b. Improved abusive comment moderation with user embeddings. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 51–55, Copenhagen, Denmark. Association for Computational Linguistics.

Andraž Pelicon, Ravi Shekhar, Blaž Škrlj, Matthew Purver, and Senja Pollak. 2021. Investigating cross-lingual training for offensive language detection. *PeerJ Computer Science*, 7:e559.

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. 2020. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, pages 1–47.

Senja Pollak, Marko Robnik-Šikonja, Matthew Purver, Michele Boggia, Ravi Shekhar, Marko Pranjić, Salla Salmela, Ivar Krustok, Tarmo Paju, Carl-Gustav Linden, Leo Leppänen, Elaine Zosa, Matej Ulčar, Linda Freienthal, Silver Traat, Luis Adrián Cabrera-Diego, Matej Martinc, Nada Lavrač, Blaž Škrlj, Martin Žnidaršič, Andraž Pelicon, Boshko Koloski, Vid Podpečan, Janez Kranjc, Shane Sheehan, Emanuela Boros, Jose G. Moreno, Antoine Doucet, and Hannu Toivonen. 2021. EMBEDDIA tools, datasets and challenges: Resources and hackathon contributions. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*, pages 99–109, Online. Association for Computational Linguistics.

Cristian Popa and Traian Rebedea. 2021. Bart-tl: Weakly-supervised topic label generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1418–1425.

Dietmar Schabus and Marcin Skowron. 2018. Academic-industrial perspective on the development and deployment of a moderation system for a newspaper website. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Dietmar Schabus, Marcin Skowron, and Martin Trapp. 2017. One million posts: A data set of german online discussions. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1241–1244.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

Ravi Shekhar, Marko Pranjić, Senja Pollak, Andraž Pelicon, and Matthew Purver. 2020. Automating News Comment Moderation with Limited Resources: Benchmarking in Croatian and Estonian. *Journal for Language Technology and Computational Linguistics (JLCL)*, 34(1).

Jan Šnajder. 2014. DerivBase.hr: A high-coverage derivational morphology resource for Croatian. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3371–3377, Reykjavik, Iceland. European Language Resources Association (ELRA).

Andrej Švec, Matúš Pikuliak, Marián Šimko, and Mária Bieliková. 2018. Improving moderation of online discussions via interpretable neural models. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 60–65, Brussels, Belgium. Association for Computational Linguistics.

Fei Tan, Yifan Hu, Changwei Hu, Keqian Li, and Kevin Yen. 2020. TNT: Text normalization based pre-training of transformers for content moderation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4735–4741, Online. Association for Computational Linguistics.

Thanh Tran, Yifan Hu, Changwei Hu, Kevin Yen, Fei Tan, Kyumin Lee, and Se Rim Park. 2020. HABERTOR: An efficient and effective deep hatespeech detector. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7486–7502, Online. Association for Computational Linguistics.

Bertie Vidgen and Leon Derczynski. 2020. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *Plos one*, 15(12):e0243300.

Wenlin Wang, Zhe Gan, Wenqi Wang, Dinghan Shen, Jiaji Huang, Wei Ping, Sanjeev Satheesh, and Lawrence Carin. 2018. Topic compositional neural language model. In *International Conference on Artificial Intelligence and Statistics*, pages 356–365. PMLR.