Cross-lingual Capsule Network for Hate Speech Detection in Social Media

Aiqi Jiang
Queen Mary University of London
London, UK
a.jiang@qmul.ac.uk

Arkaitz Zubiaga
Queen Mary University of London
London, UK
a.zubiaga@qmul.ac.uk

ABSTRACT
Most hate speech detection research focuses on a single language, generally English, which limits their generalisability to other languages. In this paper we investigate the cross-lingual hate speech detection task, tackling the problem by adapting the hate speech resources from one language to another. We propose a cross-lingual capsule network learning model coupled with extra domain-specific lexical semantics for hate speech (CCNL-Ex). Our model achieves state-of-the-art performance on benchmark datasets from AMI@Evalita2018 and AMI@Ibereval2018 involving three languages: English, Spanish and Italian, outperforming state-of-the-art baselines on all six language pairs.

CCS CONCEPTS
- Information systems → Social networks; Data analytics;
- Computing methodologies → Natural language processing;
- Social and professional topics → Women.

KEYWORDS
Cross-lingual Learning; Capsule Network; Hate Speech Detection; Social Media

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

HT ’21, August 30-September 2, 2021, Virtual Event, Ireland.
© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-8551-0/21/08...$15.00
https://doi.org/10.1145/3465336.3475102

1 INTRODUCTION
Anonymity and lack of moderation provide benefits for social media, alongside negative effects such as production of harmful and hateful contents [18, 32, 39]. The necessity to tackle hate speech has attracted the attention of the scientific community, industry and government to come up with automated solutions. Most existing research focuses on English hate speech, due to its advantageous position over other languages in terms of available resources. This in turn leads to a dearth of research in other languages [18]. Despite the recent trend of increasingly investigating hate speech detection in other languages [36], most of the work is limited to a single language. As far as we know, a few documented efforts have been made on cross-lingual hate speech detection in terms of multilingual word embeddings [1, 27, 28], multilingual multi-aspect hate speech analysis [26], and state-of-the-art cross-lingual contextual embeddings like multilingual BERT [14] and XLM-RoBERTa [19, 29]. To further research in broadening the generalisability and suitability of models across languages for hate speech detection [39], we propose a Cross-lingual Capsule Network Learning model with Extra lexical semantics specifically for hate speech (CCNL-Ex), whose two-parallel framework enriches inputs information in both source and target languages. The model can be applied to new languages lacking annotated training data [7, 28] and is able to capture spatial positional relationships between words to improve the generalisability of capsules. It can also be exploited to broaden the detection capacity on linguistically diverse genres such as social media. Our key contributions include: (1) we introduce the first approach to cross-lingual hate speech detection that incorporates capsule networks; (2) we integrate hate-related lexicons into pre-trained word embeddings to investigate their potential to further boost performance; (3) our model yields state-of-the-art performance for all six language pairs under study compared with ten baselines; (4) we perform a comparative study looking into the impact of each layer on our model.

2 RELATED WORK
Cross-lingual Hate Speech Detection. With the prevalence of online social media, a range of NLP approaches, especially transformer-based techniques [35], have been employed to identify online hate speech [1, 18, 39] or focusing on detecting specific types of hate, such as racism [13, 37], sexism [27, 38], and cyberbullying [5, 30], but limited to a single language, generally in English. While more mature NLP fields such as sentiment analysis have accumulated substantial efforts by employing cross-lingual learning techniques, work on hate speech detection in cross-lingual scenario has not been explored as much. Basile and Rubagotti [3] use SVM with n-grams to tackle English and Italian in a cross-lingual setting in Evalita 2018 and achieve the 15th/2nd position for English/Italian. Pamungkas and Patti [28] propose a joint-learning cross-lingual model with multilingual HurtLex [4] and MUSE embeddings [24], which outperforms other models using monolingual embeddings [28]. Several multilingual multi-aspect approaches are conducted for hate speech [26] and cross-lingual contextual word embeddings are applied in offensive language identification from English to other languages [19, 29]. Due to the scarcity of cross-lingual resources in this field, some studies tend to generate parallel corpora directly leveraging machine translation resources such as Google Translate [9, 27, 28, 41], which has proved the effectiveness of the
approach. In our study, we consider Pamungkas and Patti [28]'s cross-lingual joint model as a baseline which also uses machine translation as part of their pipeline.

**Capsule Networks.** Capsule Network is a clustering-like method proposed by Sabour et al. [31]. They replace scalar-output feature detectors of CNNs with vector-output capsules to learn spatial relationships of entities via dynamic routing, improving representations against CNNs. Hinton et al. [22] then propose a new EM-based iterative routing, which shows potential in image analysis [31] and is soon applied to NLP research [40]. Its use on hate speech detection is however limited [33, 34]. Srivastava et al. [34] put forward a cross-lingual capsule network learning model with multilingual relationships of entities via dynamic routing, improving representational effectiveness, and attach less significance of unrelated or useless content, reduce the loss to make the capsule network more formative and attach less significance of unrelated or useless content.

3 OUR PROPOSED METHOD: CCNL-EX

3.1 Model Architecture

Inspired by Capsule Networks built by Sabour et al. [31], we propose a cross-lingual capsule network learning model with multilingual word embeddings integrated lexical semantics (CCNL-Ex). It is composed of six layers (see Figure 1):

**Input Layer.** CCNL-Ex has two parallel capsule-based architectures for bilingual input training data — $O_s$ in the source language and the parallel translated $T_t$ in the target language.

**Embedding Layer.** The input data is the sequence of texts and each text consists of a series of words. The input representation is a weight matrix $X \in \mathbb{R}^{e \times V}$ for $e$-dimensional vector of words and vocabulary size of $V$, fine-tuned by absorbing extra hate-related lexical semantic information (see Section 3.2).

**Feature Extraction Layer.** In each aligned network, we use a Bidirectional Long Short Term Memory (BiLSTM) network [21] as the feature extractor to get contextual relationships from local features. The output of BiLSTM is $h_t = [h_t^f, h_t^b] \in \mathbb{R}^{(2xk)}$, combined by forward feature $h_t^f$ and backward feature $h_t^b$ with $k$ units.

**Capsule Layer.** The capsule layer consists of a primary capsule layer and a convolutional capsule layer. The primary capsule layer extracts instantiation parameters to represent spatial position relationships between features, like local order of words and their semantics [40]. Suppose $W \in \mathbb{R}^{(2xk)xk}$ is a shared matrix, where $d$ is the dimensionality of capsules. For the hidden feature $h_t$, we create each capsule $p_s \in \mathbb{R}^d$:

$$p_s = g(W^T h_t + b)$$  \hspace{1cm} (1)

where $g$ is a non-linear squash function to compress the vector length between 0 and 1:

$$g(s) = \frac{||s||^2}{1 + ||s||^2} \frac{s}{||s||}$$  \hspace{1cm} (2)

The convolutional capsule layer is connected to capsules in the primary capsule layer. Some primitive routing algorithms, like max pooling in CNNs, only capture features to show whether it exists in a certain position or not, missing more spatial relationships [40]. The connection weight is learnt by a dynamic routing, which can reduce the loss to make the capsule network more formative and effective, and attach less significance of unrelated or useless content, such as stop words [34]. The process of dynamic routing between the primary capsule $u_i$ and the convolutional capsule $v_j$ is as below:

$$c_{ji} = softmax(b_{ji})$$  \hspace{1cm} (3)

$$v_j = g(\sum_i c_{ji} u_{ji})$$  \hspace{1cm} (4)

$$b_{ji} = b_{ji} + u_{ji} \cdot v_j$$  \hspace{1cm} (5)

where $b_{ji}$ denotes the connection weight between capsules. After the routing, all output capsules are flattened.

Figure 1: The architecture of CCNL-Ex.
Output Layer. The final representations from the two parallel architectures are concatenated, using a softmax function to obtain the label probability.

3.2 Lexical Semantic Knowledge Infusion
We fine-tune pre-trained word embeddings by infusing domain-specific lexical semantic knowledge, aiming to obtain domain-aware word representations and enhance model capacity of identifying hate-related content. More specifically, we firstly retrieve five most relevant semantic words from SenticNet [6] for each lexical word, and utilise Fasttext embedding model [20] to generate the five most similar words for each out-of-vocabulary (OOV) word. Then we apply the similarity learning method proposed by Faruqui et al. [15] to integrate lexicon-derived semantic information into pre-trained word embeddings by minimising distances between a word and its semantically related words.

4 EXPERIMENTS
We investigate cross-lingual hate speech detection as a binary classification task in three different languages – English (EN), Italian (IT) and Spanish (ES) – and all six possible language pairs involving them: ES→EN, EN→ES, IT→EN, EN→IT, ES→IT, and IT→ES.

4.1 Datasets
We use gender-based hate speech datasets from the Automatic Misogyny Identification (AMI) tasks held at the Evalita 2018\(^1\) and IberEval 2018\(^2\) evaluation campaigns. These datasets provided by AMI@Evalita and AMI@IberEval are extracted from the Twitter platform, and constructed under the same annotation scheme for binary labels: misogynistic and non-misogynistic. AMI@Evalita datasets present texts in English and Italian [16], while the AMI@IberEval ones are in Spanish and English [17]. We utilise English data only from AMI@Evalita to make data size balanced among three languages, as well as for consistency with previous research [28] to enable direct comparison. In view of the well-divided training and test sets for each language, we further randomly select 20% of the training set as the validation set for model fine-tuning process, and utilise Fasttext embedding model [20] to generate the five most similar words for each out-of-vocabulary (OOV) word. Then we apply the similarity learning method proposed by Faruqui et al. [15] to integrate lexicon-derived semantic information into pre-trained word embeddings by minimising distances between a word and its semantically related words.

Table 1: Distribution of train, validation and test sets, misogynistic text rate (MTR) in source training and test sets, data sources for three languages.

<table>
<thead>
<tr>
<th>Language</th>
<th>English (EN)</th>
<th>Spanish (ES)</th>
<th>Italian (IT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>3200</td>
<td>2646</td>
<td>3200</td>
</tr>
<tr>
<td>Validation</td>
<td>800</td>
<td>661</td>
<td>800</td>
</tr>
<tr>
<td>Test</td>
<td>1000</td>
<td>831</td>
<td>1000</td>
</tr>
<tr>
<td>MTR(_{\text{train}}) (%)</td>
<td>44.6</td>
<td>49.9</td>
<td>45.7</td>
</tr>
<tr>
<td>MTR(_{\text{test}}) (%)</td>
<td>46.0</td>
<td>49.9</td>
<td>50.9</td>
</tr>
<tr>
<td>Source</td>
<td>Evalita2018</td>
<td>IberEval2018</td>
<td>Evalita2018</td>
</tr>
</tbody>
</table>

4.2 Multilingual Lexicons
To further assess model performance, we integrate two multilingual domain-related lexicons as extended knowledge into embeddings to explore the possibility of fine-tuning word embeddings and investigate their potential to further boost performance:

HurtLex\(^3\) It is a multilingual hate speech lexicon, containing offensive, aggressive, and hateful words or phrases in over 50 languages and 17 categories. We obtain 6,287 words for English, 3,565 for Spanish and 4,286 for Italian from it.

Multilingual Sentiment Lexicon\(^4\) Since hate speech often expresses more negative sentiments [25], we utilise a sentiment lexicon, which consists of positive and negative words in 136 languages, and provides 2,955 negative words for English, 2,720 for Spanish and 2,893 for Italian [8].

4.3 Baselines
We compare both CCNL and CCNL-Ex (with lexicons infused) with ten baselines, including SVM with unigrams features, CNN, BiLSTM and CapsNet with monolingual Fasttext embeddings of translated target data, multilingual embeddings MUSE and LASER fed to a 2-layer feedforward neural network, the state-of-the-art cross-lingual models mBERT and XLM-R covered by a 2-layer feedforward classifier on the output layer, and hate-specific cross-lingual model JL-HL with two inputs proposed by Pamungkas and Patti [28]. All baselines are described as follows:

Majority. The majority classifier always predicts the most frequent class in the training set. MTR\(_{\text{train}}\) values for three languages are less than 50%, which means the majority class is non-misogynistic.

SVM. Support Vector Machine (SVM) aims to determine the best decision boundary between vectors that belong to a given category or not [12];

CNN. Consisting of one convolutional layer and one max pooling layer to capture local textual features [23];

BiLSTM. Composed of forward/backward recurrent neural networks to extract long-term dependencies of a text [10];

CapsNet. A single capsule network [31] using a convolutional layer to extract n-gram features;

LASER. Language-Agnostic SEntence Representations (LASER) aims to calculate and use joint multilingual sentence embeddings across 93 languages [2];

MUSE. Multilingual Unsupervised and Supervised Embeddings (MUSE) builds bilingual dictionaries and aligns monolingual word embedding spaces without supervision [24];

mBERT. Multilingual BERT\(^5\) (mBERT) is a variant of BERT [14] that was trained on 104 languages of Wikipedia;

XLM-R. XLM-RoBERTa is a scaled cross-lingual sentence encoder across 100 languages from Common Crawl [11];

---

\(^1\)https://amiievalita2018.wordpress.com/data/
\(^2\)https://amiievalita2018.wordpress.com/important-dates/data/
\(^3\)https://translate.google.co.uk/
\(^4\)http://hatespeech.unito.it/resources.html
\(^5\)https://sites.google.com/site/datascienceslab/projects/multilingualsentiment
\(^6\)https://github.com/google-research/bert/blob/master/multilingual.md
Table 2: Comparison of CCNL and CCNL-Ex over baselines on the six language pairs. The best result in bold and the second best result underlined.

<table>
<thead>
<tr>
<th>Model</th>
<th>ES→EN</th>
<th>EN→ES</th>
<th>IT→EN</th>
<th>EN→IT</th>
<th>ES→IT</th>
<th>IT→ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Majoriry</td>
<td>0.351</td>
<td>0.334</td>
<td>0.351</td>
<td>0.329</td>
<td>0.329</td>
<td>0.334</td>
</tr>
<tr>
<td>SVM</td>
<td>0.620</td>
<td>0.561</td>
<td>0.588</td>
<td>0.227</td>
<td>0.643</td>
<td>0.525</td>
</tr>
<tr>
<td>CNN</td>
<td>0.598</td>
<td>0.613</td>
<td>0.592</td>
<td>0.275</td>
<td>0.636</td>
<td>0.607</td>
</tr>
<tr>
<td>BiLSTM</td>
<td>0.575</td>
<td>0.608</td>
<td>0.597</td>
<td>0.341</td>
<td>0.498</td>
<td>0.459</td>
</tr>
<tr>
<td>CapsNet</td>
<td>0.616</td>
<td>0.559</td>
<td>0.601</td>
<td>0.323</td>
<td>0.555</td>
<td>0.611</td>
</tr>
<tr>
<td>LASER</td>
<td>0.552</td>
<td>0.466</td>
<td>0.597</td>
<td>0.374</td>
<td>0.678</td>
<td>0.619</td>
</tr>
<tr>
<td>MUSE</td>
<td>0.592</td>
<td>0.491</td>
<td>0.618</td>
<td>0.400</td>
<td>0.717</td>
<td>0.666</td>
</tr>
<tr>
<td>mBERT</td>
<td>0.567</td>
<td>0.580</td>
<td>0.568</td>
<td>0.399</td>
<td>0.648</td>
<td>0.618</td>
</tr>
<tr>
<td>XLM-R</td>
<td>0.583</td>
<td>0.618</td>
<td>0.597</td>
<td>0.411</td>
<td>0.677</td>
<td>0.613</td>
</tr>
<tr>
<td>JL-HL</td>
<td>0.635</td>
<td>0.687</td>
<td>0.605</td>
<td>0.497</td>
<td>0.660</td>
<td>0.637</td>
</tr>
<tr>
<td>CCNL</td>
<td>0.624</td>
<td>0.719</td>
<td>0.628</td>
<td>0.584</td>
<td>0.735</td>
<td>0.668</td>
</tr>
<tr>
<td>CCNL-Ex</td>
<td>0.651</td>
<td>0.729</td>
<td>0.629</td>
<td>0.519</td>
<td>0.736</td>
<td>0.670</td>
</tr>
</tbody>
</table>

**JL-HL.** A joint-learning cross-lingual model proposed by Pamungkas and Patti [28], a hybrid approach with LSTM architectures which concatenates multilingual lexical features. The source data and translated target data are fed to two parallels separately.

### 4.4 Experiment Settings

For training our model, we use FastText embeddings of dimension 300 trained on the Common Crawl and Wikipedia [20]. We use 128 units for forward and backward LSTMs (256 units in total) and 50 units in hidden layer for the feedforward classifier. For capsule networks, we use 10 capsules of dimension 16 and the number of dynamic routing is 5. We use Adam optimiser with 0.0001 learning rate, and set 0.4 for dropout value and 8 for batch size. The model is coded in Keras 2.2.4 and Tensorflow 1.14. We run experiments on the HPC resources of our university, each experiment taking less than one hour. Macro-averaged F1 score is reported as the evaluation metric for all experiments.

### 5 RESULTS

#### 5.1 Model Performance

Results are shown in Table 2. CCNL and CCNL-Ex differ in that the latter incorporates lexical semantic features. We can observe that CCNL yields better performance than all baseline models for five out of six language pairs, with the exception of ES→EN. CCNL-Ex outperforms all ten baselines for all language pairs. These results substantiate the effectiveness of our model with semantic information, highlighting its generalisation capability across three languages.

Among the ten baselines, the best is JL-HL, whose performance is still always below that of CCNL-Ex. CCNL achieves absolute improvements ranging 7%-9% over JL-HL model for two language pairs involving Italian: EN→IT, and ES→IT. In addition, the CCNL model has manifested pronounced improvements in terms of separately identifying two classes compared to the majority class baseline. We can also observe that CCNL generally achieves a large margin with respect to other baselines like SVM, CNN and BiLSTM (especially for ES→EN, EN→IT and IT→ES), while the baseline MUSE achieves good results for IT→EN and IT→ES. It also highlights the effectiveness of the capsule network as an important component in the cross-lingual model compared with other baselines. Furthermore, we observe that CCNL achieves better performance on all six language pairs when we compare it with CapsNet, LASER, mBERT and XLM-R. Possible reasons are that BiLSTM layers enable the proposed model the capability of extracting contextual information compared to the CNN layer, and CCNL takes the spatial features into consideration by sharing the same weight matrix and learning the positional feature difference in high level via the dynamic routing process.

Compared with CCNL, CCNL-Ex performs better for ES→EN and EN→ES, and shows a similar performance in three out of six language pairs, which indicates the effectiveness of integrating semantics based on lexicons. The exception to the trend showing a better performance for lexicon-based methods is for the two language pairs that have Italian as the target, namely EN→IT and ES→IT, where the base CCNL model with no lexicons performs best. This is likely due to limitations in the Italian language lexicons, and hence reinforces the need to secure high quality lexicons if they are to be incorporated.

#### 5.2 Comparative Experiments

In order to explore the effect of diverse components in our cross-lingual capsule model on six language-pair tasks, we further implement experiments to assess ablated models compared to our basic framework CCNL and the impact of varying specific components in the feature extraction layer.

5.2.1 Framework Ablation Analysis. We perform an ablation study for CCNL by dropping one of the two parallel architectures (CCNL-non-parallel), removing the LSTM layer (CCNL-non-LSTM) and removing the Capsule Network layer (CCNL-non-Caps). As shown in Table 3, CCNL outperforms all ablated models, demonstrating the combined benefits of all CCNL components. CCNL noticeably outperforms CCNL-non-parallel on all language pairs, highlighting the importance of the two-parallel framework for extracting local features from both source and target texts. Additionally, we can validate the ability of the BiLSTM network to
Table 3: CCNL comparative experiment results for six language pairs. The best result in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>ES→EN</th>
<th>EN→ES</th>
<th>IT→EN</th>
<th>EN→IT</th>
<th>ES→IT</th>
<th>IT→ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Results for ablation experiments</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCNL-non-parallel</td>
<td>0.522</td>
<td>0.558</td>
<td>0.570</td>
<td>0.513</td>
<td>0.626</td>
<td>0.624</td>
</tr>
<tr>
<td>CCNL-non-LSTM</td>
<td>0.373</td>
<td>0.609</td>
<td>0.565</td>
<td>0.406</td>
<td>0.685</td>
<td>0.623</td>
</tr>
<tr>
<td>CCNL-non-Caps</td>
<td>0.597</td>
<td>0.678</td>
<td>0.613</td>
<td>0.439</td>
<td>0.643</td>
<td>0.622</td>
</tr>
<tr>
<td>CCNL</td>
<td><strong>0.624</strong></td>
<td>0.719</td>
<td><strong>0.628</strong></td>
<td><strong>0.584</strong></td>
<td><strong>0.737</strong></td>
<td><strong>0.668</strong></td>
</tr>
<tr>
<td>Results for feature layers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCNL-non-FE</td>
<td>0.373</td>
<td>0.609</td>
<td>0.565</td>
<td>0.406</td>
<td>0.685</td>
<td>0.623</td>
</tr>
<tr>
<td>CCNL-CNN</td>
<td>0.521</td>
<td>0.592</td>
<td>0.577</td>
<td>0.439</td>
<td>0.633</td>
<td>0.622</td>
</tr>
<tr>
<td>CCNL-GRU</td>
<td>0.458</td>
<td><strong>0.722</strong></td>
<td>0.613</td>
<td>0.411</td>
<td>0.715</td>
<td><strong>0.671</strong></td>
</tr>
<tr>
<td>CCNL</td>
<td><strong>0.624</strong></td>
<td>0.719</td>
<td><strong>0.628</strong></td>
<td><strong>0.584</strong></td>
<td><strong>0.737</strong></td>
<td><strong>0.668</strong></td>
</tr>
</tbody>
</table>

Table 4: Examples for error analysis. Translated texts are presented for non-English instances. Ground truth (GT) and prediction (P) labels are noted – hateful (1) and non-hateful (0), along with corresponding error types (ET).

<table>
<thead>
<tr>
<th>Text</th>
<th>GT</th>
<th>P</th>
<th>ET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analicemos esto: ¿Si te pones unos shorts así, en la calle, ¿qué esperas que te digan? ¿Acoso? ¿O Provocación...</td>
<td>1</td>
<td>0</td>
<td>a</td>
</tr>
<tr>
<td>Translation: Let’s analyse this: If you wear shorts like this, in the street, what do you expect them to say?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bullying? Or Provocation...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tranquille ragazze, tranquilli gay, il Butturini c’ha una morosa che un pezzo di fica mostruosa! #TVOI</td>
<td>0</td>
<td>1</td>
<td>b</td>
</tr>
<tr>
<td>Translation: quiet girls, quiet gays. Butturini has a girlfriend who is a piece of monstrous pussy! #TVOI</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>@user ben sasse is 100% correct. since 1973, all ive ever heard every two years for elections are hysterical women (all a leftist act) about back-alley abortions. this shit is getting old! i didn’t hear one other protest issue being yelled about i</td>
<td>1</td>
<td>0</td>
<td>c</td>
</tr>
<tr>
<td>@user ma se la #culona #tedesca che predica #austerit mi sono perso qualcosa</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Translation: @user but if the #culona #german preaching #austerit I missed something</td>
<td>1</td>
<td>0</td>
<td>d</td>
</tr>
</tbody>
</table>

extract contextual information effectively compared with CCNL-non-LSTM, which highlights the effectiveness of the capsule network compared with CCNL-non-Caps.

5.2.2 Impact of Feature Extraction Layer. We aim to validate the ability of the BiLSTM network to extract contextual information effectively. We test different feature extraction layers by keeping other components of the CCNL architecture unchanged. We test four flavours of CCNL: CCNL (with LSTM feature extraction layer), CCNL-non-FE (without feature extraction layer), CCNL-CNN (CNN feature extraction instead) and CCNL-GRU (bidirectional GRU feature extraction instead). Results in Table 3 show that CCNL with the feature extraction layer performs consistently better than those without it, highlighting the importance of extracting local features from the text. Additionally, CCNL also shows improved performance on all tasks when compared with CCNL-CNN, since contextual information plays a significant role in detecting hate speech. CCNL noticeably outperforms CCNL-GRU on four out of six language pairs and achieves similar performance (differences below 0.5%) on the other two language pairs (EN→ES and IT→ES). This is likely due to structural similarities of GRU and LSTM, with the additional complexity of LSTM allowing to capture more informative features in some language pairs.

5.3 Error Analysis

We inspect frequent errors across misclassifications from the test set by the CCNL-Ex model (see Table 4 for examples). We summarise the following four main types of errors:

(a) Implicit hate: Those lacking explicit hateful content or context in the post;

(b) Overuse of hateful words: Hateful words can be overused, leading to the over-dependence of the model on these words, while hate targets in posts are confounding and hard to be identified;

(c) Lack of prior information: The model cannot identify those contents referring to hate-related event, people or words/phrases with special meanings as it does not possess prior knowledge;

(d) Erroneous translation: The use of machine translation can lead to translation errors for important words. Some words used in hashtags cannot be easily translated, which might be regarded as out-of-vocabulary words by the model.

6 CONCLUSION AND FUTURE WORK

We propose a Cross-lingual Capsule Network Learning model integrating Extra hate-related semantic features (CCNL-Ex) for hate speech detection. CCNL, main framework of our model, is composed of two parallel architectures for source and target languages, using BiLSTM to extract contextual features and Capsule Network to capture hierarchically positional relationships. Our model finally leads to state-of-the-art performance for all six language pairs compared with ten competitive baselines. Results show the potential of learning contextual information and spatial relationships of hate speech texts. Given that using machine translation resources such as Google translate is not always a perfect option with very informal language such as those used on social media for hate speech, we will explore approaches to consider culturally grounded context
Aiqi Jiang is funded by China Scholarship Council (CSC). This research utilised Queen Mary’s Apocrita HPC facility, supported by QMUL Research-IT. http://doi.org/10.5281/zenodo.438049

ACKNOWLEDGMENTS

Aiyé Arango, Jorge Pérez, and Barbara Bobleto. 2020. Hate speech detection is not as easy as you may think: A closer look at model validation (extended version).


Elisabetta Fersini, Paolo Nozza, and Paolo Rosso. 2018. Overview of the Evalita 2018 Task on Automatic Misogyny Identification (AM). In EVALITA@ CIcL-it.


REFERENCES