Heidelberg University
Department of Computational Linguistics

COMPARISON OF MULTILINGUAL WORD AND SENTENCE EMBEDDINGS FOR CROSS-LINGUAL CRISIS RECOGNITION

Bachelor Thesis

Bachelor of Arts
Computational Linguistics

by

Alina Klerings
klerings@cl.uni-heidelberg.de
Matriculation number: 3478999

Supervisor Prof. Dr. Katja Markert
Reviewer Prof. Dr. Michael Herweg

Date of Submission: September 26, 2020

Declaration of Authorship

I hereby certify that the work presented here is, to the best of my knowledge and belief, original and the result of my own investigations, except as acknowledged, and has not been submitted, either in part or whole, for a degree at this or any other university.

Eigenständigkeitserklärung

Hiermit versichere ich, die vorliegende Abschlussarbeit selbstständig und nur unter Verwendung der von mir angegebenen Quellen und Hilfsmittel verfasst zu haben. Sowohl inhaltlich als auch wörtlich entnommene Inhalte wurden als solche kenntlich gemacht. Die Arbeit hat in dieser oder vergleichbarer Form noch keinem anderem Prüfungsgremium vorgelegen.

Heidelberg, September 26, 2020 

Abstract

During natural disasters and other mass emergencies, affected individuals and first responders often share valuable live information about damaged infrastructure, endangered people and offers of help via social media platforms. The efforts of concerned authorities to separate informative posts from expressions of sympathy and completely unrelated content can be supported with automatic content filtering using machine learning. This bachelor thesis compares different approaches for binary informativeness classification of crisis posts. As crisis-relevant content may not be limited to one language, the focus of this work lays on cross-lingual classification, for which the transfer ability of models across languages is evaluated. Based on a consolidated multilingual dataset of crisis tweets in English, Spanish and Italian, different word and sentence embeddings are leveraged as features for non-neural and neural architectures to separate informative from non-informative tweets. The experiments include a zero-shot scenario as well as two setups using machine translation. The comparison between context-less word embeddings (word2vec), contextualised word embeddings (BERT) and language-agnostic sentence embeddings (LASER) shows, that LASER is the best strategy for zero-shot. If translation is available, context-less word embeddings compete with the more time-intensive BERT approach. Furthermore, a quantitative analysis of the results indicates that tweet features like the occurrence of URLs and Twitter-specific user mentions positively affect the performance of all models, just as the increased length of a post. The thesis provides a comparative analysis of three different classification approaches based on one consolidated tweet corpus. Thereby, it joins the progress of related work, that has been conducted using differing crisis datasets.

Zusammenfassung

Während Naturkatastrophen und anderen Ausnahmezuständen teilen betroffene Personen und Ersthelfer in sozialen Netzwerken häufig nützliche Informationen zu zerstörten Infrastrukturen, gefährdeten Personen und Hilfsangeboten. Solche informativen Beiträge müssen jedoch von Beileidsbekundungen und nicht relevanten Posts getrennt werden. Zuständige Institutionen wie Regierungen oder humanitäre Hilfsorganisationen können dabei durch automatische Filter auf Basis von maschinellem Lernen unterstützt werden. In dieser Bachelorarbeit werden verschiedene Ansätze für eine solche binäre Klassifikation von Krisenbeiträgen bezüglich ihres Informationsgehalts verglichen. Der Fokus liegt dabei auf sprachübergreifender Klassifikation, bei der Modelle auf einer Sprache trainiert und auf einer anderen Sprache evaluiert werden. Anhand eines gemischten Datensatzes, bestehend aus englischen, spanischen und italienischen Krisen-Tweets, werden verschiedene Textrepräsentationen auf Wort- und Satzebene verglichen, indem sie als Features für nicht-neurale und neurale Architekturen verwendet werden. Dabei werden sowohl Experimente mit maschineller Übersetzung als auch ohne Übersetzung durchgeführt. Der Vergleich von kontextlosen Wortrepräsentationen (word2vec), kontextualisierten Wortrepräsentationen (BERT) und sprachunabhängigen Satzrepräsentationen (LASER) zeigt, dass bei Versuchen ohne Übersetzung letztere am erfolgreichsten die Tweets abbilden. Bei der Einbindung von Übersetzung konkurrieren kontextlose Wortrepräsentationen mit dem zeitaufwendigeren BERT Ansatz. Des Weiteren gibt eine quantitative Analyse der Ergebnisse Hinweise darauf, dass bestimmte Eigenschaften eines Tweets, wie das Vorhandensein von URLs und Twitter-spezifischen Usermentions sowie eine größere Textlänge, sich positiv auf die Leistung aller Modelle auswirken. Diese Arbeit führt einen Vergleich von drei verschiedenen Klassifizierungsansätzen anhand eines einheitlichen Tweetkorpus durch und bringt somit verwandete Forschungansätze zusammen, die bisher nur auf verschienden Datensets evaluiert wurden.

Contents

List of Figures

List of Tables

1 Introduction

In this thesis, I will conduct a comparative study on state-of-the-art cross-lingual text classification techniques using multi-purpose representations as word2vec embeddings, multilingual BERT embeddings and Language-Agnostic Sentence Representations (LASER) to distinguish informative tweets from uninformative content in cases of crises.

Natural disasters and other mass emergencies like terror attacks, industrial accidents, building collapses or train crashes can severely impact local communities and economies and entail global consequences. Therefore, well structured and networked disaster management is crucial. Social media platforms like the microblogging service Twitter have changed the broadcasting of news fundamentally and thus, opened new information retrieval opportunities. With smartphones and other smart mobile devices as our daily companions, individuals are able to retrieve and share live information about unusual events within seconds. People at the actual scene of event can help to build accurate situational awareness by assessing damage and victims. This unofficial information from first responders can prove extremely valuable. It includes, in particular but not exclusively, posts about blocked roads, flooded areas and endangered or injured individuals. Decision-makers and humanitarian aid organisations can leverage this timely relevant information about fatalities and damaged infrastructures by streaming posts in social media networks to get an overview of the situation and take respective measures. However, one challenging issue that confronts concerned authorities is the mass of content that is being posted every second. Simply monitoring the data is not enough. In order to harness it, a rough filtering followed by a more fine-grained categorisation is required. This means that disaster-related posts are separated from unrelated ones and later on, informative crisis content is distinguished from non-informative crisis content which often consists of expressions of sympathy by social media users all over the world.

Since manual filtering is not feasible to the required extent, machine learning models for information retrieval and text classification are used. These require large amounts of train data. As mentioned above, not all disaster-related information is broadcasted in the local language. For instance, news agencies from other countries commonly summarise current events with short delay and thus, provide additional sources of information. Therefore, models need to be capable of processing posts in several languages. This can become an issue, since sufficient train data is not available for all languages. Especially languages of some developing countries, where natural disasters usually hit the hardest, do not have corresponding text

corpora on which machine learning models could be trained. Collecting the required train data when a crisis has already happened is not possible because it is necessary to take immediate response actions.

A solution are models that are able to apply their learned classification skills from one language with already existing large corpora (like English) to another language with little to no training resources. In Computational Linguistics this process is known as cross-lingual transfer learning for text classification (see Section 2.1).

This thesis will compare state-of-the-art language-agnostic and multilingual word and sentence embeddings on the task of cross-lingual text classification in the case of disaster tweets, which are classified according to their informativeness. While there is related work that tackles the cross-lingual challenge, one of the shortcomings is the focus on only one dataset and one approach per analysis. This work aims at creating a bigger comparison of different cross-lingual approaches to measure the progress of crisis tweet classifiers while also consolidating a multilingual crisis dataset with fused binary labels. Such a multilingual dataset is also important for further evaluation of more sophisticated models in the future. This study examines the three languages English, Spanish and Italian and the three following types of embeddings: word2vec, multilingual BERT and LASER.

The remainder of this thesis is structured as follows: First, I will take a look at existing work regarding cross-lingual text classification and crisis-related tweet classification. In the following chapter, the utilised datasets including the preprocessing procedure are presented, as well as the theory behind the different architectures that are compared in multiple experiments in Chapter 4. All results and a quantitative analysis regarding learning behaviour and tweet features are reported in the subsequent chapter, followed by an extensive discussion in Chapter 6 and a conclusion with an outlook and open questions in Chapter 7.

2 Related Work

In this chapter, I will take a look at current state-of-the-art approaches tackling cross-lingual tasks as well as existing work regarding cross-lingual crisis tweet classification.

2.1 Cross-lingual Transfer Learning

The terms transfer learning and domain adaptation are sometimes used interchangeably in the literature, however, Pan and Yang [2010, p. 3] provide a notation that defines the latter as a subcategory of the first: "Given a source domain D_S and learning task T_S, a target domain D_T and learning task T_T, transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in D_T using the knowledge in D_S and T_S, where $D_S \neq D_T$, or $T_S \neq T_T$". In the case of cross-lingual transfer learning, we deal with the same task ($T_S = T_T$) but two different language domains ($D_S \neq D_T$) which is categorised by Pan and Yang [2010] as transductive transfer learning[1].

The transfer of linguistic knowledge across languages is a promising approach to overcome a general lack of data and improve machine learning performance for low-resource languages. To facilitate this transfer and guarantee a high cross-lingual generalisation ability, multilingual data representations are employed.

Such distributed data representations, called embeddings, are vectors, that are learned from unlabelled corpora. They serve as general purpose text representations that can be integrated as features into task-specific architectures. Embeddings represent an abstraction away from a word's string and towards the actual meaning, resulting in similar embeddings for synonyms. With Continuous Bag-of-Words and Skipgram (see Section 3.2.1), Mikolov et al. [2013a] and Mikolov et al. [2013b] introduce two models that allow efficient training of language models for word embeddings on huge corpora.

However, these word embeddings are not fully contextualised. This means, ambiguous words like 'play' only have one vector representation for all interpretations, regardless of the context they are used in and their relative position in a sentence (e.g. 'He decided to play the piano while she went to see a play in the theatre.'). The release of BERT [Devlin et al., 2018],

1 Note that the definition of Pan and Yang [2010] requires that part of the unlabelled target data is available during training which is not the case for this work.

Bidirectional Encoder Representations from Transformers, addresses this shortcoming. It facilitates big performance improvements in several language processing tasks by generating contextualised word representations, which take into account a word's position as well as its left and right context. Therefore, they are able to differentiate between different word interpretations. The detailed learning objective of BERT will be explained in Section 3.2.2. BERT not only marks a defining shift from simple word embeddings towards advanced sentence-level representations but also quickly entailed similar developments for cross-lingual tasks with multilingual embeddings. Multilingual embeddings can either be learned from parallel corpora with word, sentence or document alignment [Ruder et al., 2019] or without any cross-lingual alignment. Multilingual BERT (mBERT), which is competitive with the best cross-lingual transfer approaches [Wu and Dredze, 2019], and XLM [Lample and Conneau, 2019] are examples of such language models, that are pre-trained on corpora in multiple languages without any cross-lingual supervision. The improved XLM-R model [Conneau et al., 2019], trained on an even bigger dataset, in fact competes with strong monolingual models. Because their learned embeddings are multilingual, the models require language-specific fine-tuning of all their parameters (see Section 3.2.2).
For evaluating such fine-tuned models in a cross-lingual scenario, there are different possible setups. Besides the use of translated data – either train or test data –, common cross-lingual training setups are few-shot and zero-shot where, respectively, only a few to none examples of the target language are added to the train data in a high resource source language.
Drawbacks of fine-tuning multilingual models for different languages are, that it requires extra time and target language data (except zero-shot), and that it leads to the diversion of word vectors in different languages. For that reason, zero-shot transfer learning might be favoured over additional language-specific fine-tuning, when computational or data resources are sparse. Addressing this challenge, Artetxe and Schwenk [2019] propose LASER, language-agnostic sentence embeddings, that represent semantically similar sentences close to each other in a shared vector space, independent of their language and the faced language processing task. This means two sentences in two languages with similar meaning have a similar vector. Therefore, LASER embeddings (as opposed to BERT embeddings) can be used for cross-lingual transfer learning tasks without any target-specific fine-tuning (see Section 3.2.3).

Though huge models like XLM-R, that are trained with appropriate resources, are able to compete with single-language models for some tasks and languages, there is still a gap between monolingual and cross-lingual generalisation capabilities of deep models for the

majority of language processing tasks and languages. Hu et al. [2020] present a multilingual benchmark for nine Natural Language Processing (NLP) tasks including document classification on 40 different languages. They evaluate deep contextual representations in zero-shot and translated cross-lingual transfer settings with English as a source language. While their survey covers standard datasets such as the XNLI corpus, this thesis will conduct a similar analysis that focuses on the specific challenge of cross-lingual crisis tweet classification. This task slightly differs from common text classification due to the short nature of tweets (limit of 280 characters[2]) and the frequent use of emojis, abbreviations and slang.
This work looks exemplarily at three types of data representations: context-less word embeddings (word2vec), contextualised word embeddings (BERT) and sentence embeddings (LASER).

2.2 Cross-lingual Crisis Tweet Classification

Tweets are a subcategory of texts distinguishing themselves through short, often colloquial content. For a filtering application that is able to separate informative from non-informative crisis tweets in real time during an emergency situation, pre-trained classifiers are required that can handle multilingual input data. The work of Khare et al. [2018] and Khare et al. [2019] explores the potential of semantic knowledge from BabelNet and DBpedia added to a Support Vector Machine with features like POS, n-grams, text length and counts of hashtags and user mentions to approach the classification task. They show that feeding additional semantic features to a traditional statistic classifier brings a small improvement when translating the test data into the source language and a significant improvement for zero-shot transfer learning without translation. However, they classify according to relatedness not informativeness, which might be less useful in a real world application. Other work like Alam et al. [2020] deals with informativeness classification of crisis tweets but primarily for the English language. Khare et al. [2019] also touch upon the related task of cross-domain adaptation, which has already been emphasised as a parallel challenge to cross-lingual classification by Imran et al. [2016]. Every type of crisis has its own characteristics, so training a model on solely earthquake data may impact its generalisation ability, when applied to tweets about a hurricane event for instance. The removal of this bias towards the train data for a specific crisis is related to cross-lingual adaptation. In Imran et al.

2 https://en.wikipedia.org/wiki/Twitter#Tweets

[2016], both challenges are combined and approached with a random forest classifier learning from unigram and bigram features. Most relevant to this thesis is their finding that similar languages as Italian and Spanish are better suited for transfer learning than other pairs like Italian and English. There is further work [Lin et al., 2019], that considers the choice of the best source language for transfer learning as a ranking problem. This could be the subject of future research given sufficient train data in more languages.

Similar to monolingual classification challenges, deep learning models with general purpose embeddings for cross-lingual tasks superseded traditional non-neural approaches in the last years. Several work regarding the utilisation of language-agnostic and multilingual word embeddings with deep neural architectures like CNN [Lorini et al., 2019], [Bruijn et al., 2020] and LSTM [Torres, 2019] has been conducted in the field of cross-lingual crisis tweet classification.

However, context-aware sentence-level representations like multilingual BERT and LASER embeddings have not been explored for cross-lingual crisis classification yet.

Furthermore, previous work has brought confusion regarding the reasonable merging of categories when combining differing labelling schemes of crisis datasets in different languages, especially for binary classification. More precisely, the proper differentiation of the terms "relevance","relatedness" and "informativeness" has been overlooked in previous work, leading to unfortunate category merging for multilingual tweet datasets. An extensive analysis of the existing crisis datasets and their labels is conducted in Section 3.1.1. With a new fusion of existing labels, this thesis will compare a range of state-of-the-art cross-lingual machine and deep learning approaches for binary crisis-tweet classification.

3 Materials and Methodology

The following chapter will be split into two sections of which the first will present the data used for the analysis, including its processing. The second half will look at the theory behind the approaches to compare, namely word2vec embeddings with a Support Vector Machine, mBERT and LASER with a feed-forward neural network.

3.1 Data Processing

The process of selecting suitable datasets in respective languages, consolidating their different label categories and language-tagging, preprocessing and translating the tweets is described in this section.

3.1.1 Datasets and Labels

In order to facilitate cross-lingual transfer learning and evaluation, a multilingual crisis dataset with uniform labels is required. As mentioned in Chapter 2, previous work explores options to merge similar or close labels of datasets in different languages to evaluate multilingual classification strategies. However, since multilingual crisis data is collected separately by different researchers and is therefore labelled based on different annotation schemes, special caution has to be paid. By examining the annotation guidelines of several available datasets, I found that a binary separation of informative and non-informative tweets is the most reasonable approach based on the given labels of the multilingual data and the intended real world purpose. Distinguishing non-informative tweets, that may relate to a disaster, from content, that not only relates to a given crisis but additionally contains assessments of damage, victims and people in need of help, produces a valuable outcome for humanitarian relief operations that rely on real-time situation evaluation rather than crisis-related expressions of sympathy.
In the following, I will describe the data used for my analysis, namely three collections that are publicly available on CrisisLex, a web page that provides crisis-related social media data [Olteanu et al., 2014]. As previous work, my study looks into the trio English, Spanish

and Italian, as these languages provide sufficient annotated crisis tweets. With English being a high resource language and Italian and Spanish being linguistically close, it can be investigated whether the amount of training resources or the linguistic closeness of source and target language is more important for successful transfer learning.
The first collection, SOSItalyT4 (T4), is introduced by Cresci et al. [2015] as an Italian tweet corpus covering four different natural disasters, namely earthquakes and floods, that happened in Italy between 2009 and 2014. The tweets are collected through the Twitter Streaming API[1], then sampled by keywords and labelled by three different annotators as follows [Cresci et al., 2015, p. 2]:

- **damage:** "related to the disaster and carrying information about damages to the infrastructures or on the population"
- **no damage:** "related to the disaster but not carrying relevant information for the assessment of damages"
- **not relevant:** "not related to the disaster"

The ChileEarthquakeT1 (T1) is a labelled subset of a Spanish tweet corpus around the Chilean Earthquake in 2015 [Cobo et al., 2015]. It was constructed in the same way as T4 including keyword sampling and threefold annotation, though the labelling is slightly different [Cobo et al., 2015, p. 3]:

- **true:** related and relevant content including information about
  - "caution and advice"
  - "causalities and damage"
  - "people missing, found or seen"
  - additional "information source" of the incident
- **false:** not relevant content

1 https://developer.twitter.com/en/docs/tweets/filter-realtime/overview

The third dataset, CrisisLexT26 (T26), is significantly larger than the two preceding ones and covers 26 natural and human-induced crises between 2012 and 2013 [Olteanu et al., 2015]. It contains multiple languages, but for this analysis, only English, Spanish and Italian tweets are of interest. Again, the sampling and annotation process of this collection resembles the one of T1 and T4. Besides an informativeness label, each tweet is also annotated with regard to information type and information source which technically allows for a more fine-grained classification. This thesis however, aims at working with crisis data that only differs in language not labelling, and since T1 and T4 do not have such a fine-grained classification, the relevant tweet tags for this work are [Olteanu et al., 2015, p. 3]:

- **related and informative:** "contains useful information that helps understand the crisis situation"
- **related but not informative:** "refers to the crisis, but does not contain useful information that helps understand the situation"
- **not related**
- **not applicable / too short / not readable / other issues**

While previous work based on T1, T4 and T26 consolidated the classes according to relatedness, this thesis focuses on informativeness. The tweet examples in Table 1 give an insight into why this separation is a better choice for real world Crisis Informatics applications.

Label	**Crisis**	**Tweet**
Related and informative	Philippines Flood, 2012	RT @sakatunayan: 21 stuck at the 2nd floor of a house at 22 b evangelista st xavierville 1 qc. Pls send help! #rescuePH
Related but not informative	Boston Bombings, 2013	#prayforboston bless their hearts!
Not Related	Philippines Flood, 2012	#rescueph Free 1000 tv channels on your phone? http://t.co/W6KxbE8N http://t.co/lQuSz8YA fufu

Table 1: Example tweets and their labels from CrisisLex T26

As can be seen on the Boston Bombings tweet, uninformative expressions of sympathy are often shared during a disaster and have to be filtered out. Therefore, in the following,

tweets labelled with *damage* (T4), *true* (T1) or *related and informative* (T26) are considered informative while all other remaining tweets are tagged with a non-informative label. The resulting label distribution of the constructed multilingual dataset after all preprocessing steps is presented in Table 2. Note that the class *not applicable* from T26 was excluded, as it cannot be converged to one of the two final categories. The varying label distributions per language are addressed with train data downsampling during the experiments.
Another challenge that presents itself is the use of crisis associated hashtags with completely unrelated content to take advantage of temporarily trending keywords, as can be seen in the case of the television channel advertisement during the Phillipines flood in Table 1. To recognise this kind of deceptive tagging, the context of the whole tweet needs to be understood, therefore this analysis looks into sentence-level representations in particular.

CL	Original Label	Counts	Binary Label	Counts
		SPANISH		
T1	True	825	INFORMATIVE	1,904
T26	Related and informative	1,079		
T1	False	1,053	NON-INFORMATIVE	1,817
T26	Related but not informative	588		
T26	Not related	176		
		ITALIAN		
T4	Damage	1,555	INFORMATIVE	2,320
T26	Related and informative	765		
T4	No damage	2,400	NON-INFORMATIVE	3,708
T4	Not relevant	739		
T26	Related but not informative	483		
T26	Not related	86		
		ENGLISH		
T26	Related and informative	10,665	INFORMATIVE	10,665
T26	Related but not informative	4,186	NON-INFORMATIVE	5,799
T26	Not related	1,613		

Table 2: Consolidated multilingual dataset after label fusion and preprocessing

3.1.2 Language Identification

For the cross-lingual analysis and the comparison of different source and target languages, it is important to have clear language tags for all tweets. Therefore, all tweets of the

CrisisLex collections are categorised by means of a language detection pipeline. This is of particular importance for the mixed T26, but also for the primarily monolingual T1 and T4 which contain some tweets in other languages, too. Therefore, an ensemble of three simple off-the-shelf language identifiers is utilised as suggested in Lui and Baldwin [2014] and Jauhiainen et al. [2019]. In a first step, the language of a tweet is identified by the three tools cld2[2], langid[3] and langdectect[4]. Tweets with a threefold agreement of the classifiers are kept, while all other tweets are additionally classified by detectlanguage[5], an application with limited requests per day. If they cannot achieve a three out of four language agreement, they are discarded. While this ensemble guarantees a high confidence language identification, it only follows the majority vote and handles instances that contain code switching, a change of language within the tweet text, as monolingual. However, since I am also going to investigate the use of language-agnostic sentence representations, I decided to keep the respective tweets.

3.1.3 Preprocessing

In the next step, the tweet data is cleaned and preprocessed to improve machine readability. For all experiments with BERT and LASER this entails

- discarding all problematic tweets (labelled “not applicable”),
- removing all additional whitespaces, including newlines,
- replacing HTML character codes (&amp, &lt, &gt),
- removing "..." endings from not fully retrieved tweets[6],
- removing all occurrences of #,
- replacing URLs with UL token,
- replacing user mentions with UQ token,

2 Github: https://github.com/CLD2Owners/cld2
3 Github: https://github.com/saffsd/langid.py
4 https://pypi.org/project/langdetect/
5 https://detectlanguage.com/
6 Some retrieved tweets end with "..." instead of the real text, which is caused by incorrectly set parameters of the Twitter streaming tool. To exclude affected tweets or to request them again are two options to deal with the issue. However, if the truncated tweets were used for the annotation, the respective label is solely based on the available part. Therefore, this work keeps the affected tweets.

- unifying all retweet markers to RT,
- lowercasing (after translation),
- discarding tweets shorter than four tokens (common practice) and
- dropping duplicates.

Hashtags are commonly used on Twitter to simplify topic search and connecting users regarding a specific issue. The phrases following the hash token are equally valuable for text classification when using encodings that rely on subword tokenisation like BERT and LASER, thus they are not removed. Also, emoticons and punctuation marks are kept, as the vocabulary of the pre-trained BERT embeddings covers them. The necessary preprocessing for the generation of LASER embeddings is less documented and only includes lowercasing, according to Artetxe and Schwenk [2019], therefore I follow the same procedures as with BERT. Furthermore, for BERT, a special [CLS] token needs to be added at the beginning of every tweet to aggregate the sequence representation for classification and a [SEP] token is required in the end. The details are explained in Section 3.2.2. Because LASER and BERT embeddings are context-sensitive and are trained to deal with natural language input, no stopword removal or lemmatisation is applied. Only for generating word2vec embeddings for SVM experiments numbers, punctuation marks, emoticons and stopwords are removed and the tweets are tokenised.

3.1.4 Translation

One of the options to facilitate cross-lingual transfer learning is to translate either the train or test data into the target or source language respectively. The detailed experimental setup is explained in the next chapter but in essence, a translation of all Italian, Spanish and English tweets into the respective two other languages is required. For that purpose, the Amazon Translate service[7], a neural machine translation software, is used. With the Free Tier option, the first two million characters per month are free for one year. To save characters, all preprocessing steps except lowercasing are applied before translation.
To maintain the newly added UL, UQ and RT tokens and not accidentally translate them, they are added to custom terminology, a user-defined lexicon of words that should be treated by the Amazon Service in a certain way, in essence: special or no translation. After obtaining

7 https://aws.amazon.com/de/translate/

the translated data, all tweets are lowercased in a final preprocessing step and mapped to a numeric label of zero (non-informative) or one (informative).

3.2 Theoretical Foundations

As mentioned in the second chapter, there exists research concerning statistical and deep learning models for cross-lingual crisis tweet classification but previous work omits the direct comparison of these models in one test setting. Therefore, this analysis not only looks at advanced sentence representations for neural networks, but also implements a non-neural Support Vector Machine with context-free word embeddings as a strong baseline. The following section describes the theory behind all utilised architectures for the experiments of the thesis.

3.2.1 Word2Vec with Support Vector Machine

For the baseline, word2vec embeddings are generated for the tweet data. Word2vec [Mikolov et al., 2013a] is an algorithm for a statistical language model that learns its representations by computing the probability of an upcoming word or sentence. This section describes the training of the language model based on Mikolov et al. [2013a] and explains how it can be leveraged for text classification with a Support Vector Machine.
The unsupervised language modelling requires a large unlabelled text corpus that serves as input for a simple feed-forward neural network with one hidden layer. To create a vector space in which each corpus word is assigned one distributed word vector, there are two possible learning objectives for the network, see also Figure 1.
The first objective is to correctly predict the current word $w(t)$ given the previous and future words in the context. Because the order of the context words does not influence the projection and the average of their vectors is computed, this model is called Continuous Bag-of-Words Model (CBOW). On the other hand, there is the Skipgram strategy of predicting the context words $w(t-1), w(t-2), w(t+1), w(t+2)...$ for the current word $w(t)$. This is done by computing the probability for each word in the corpus that it is a context word of $w(t)$. In either case, the network's parameters are updated to maximise the probabilities and learn a vector space representation.

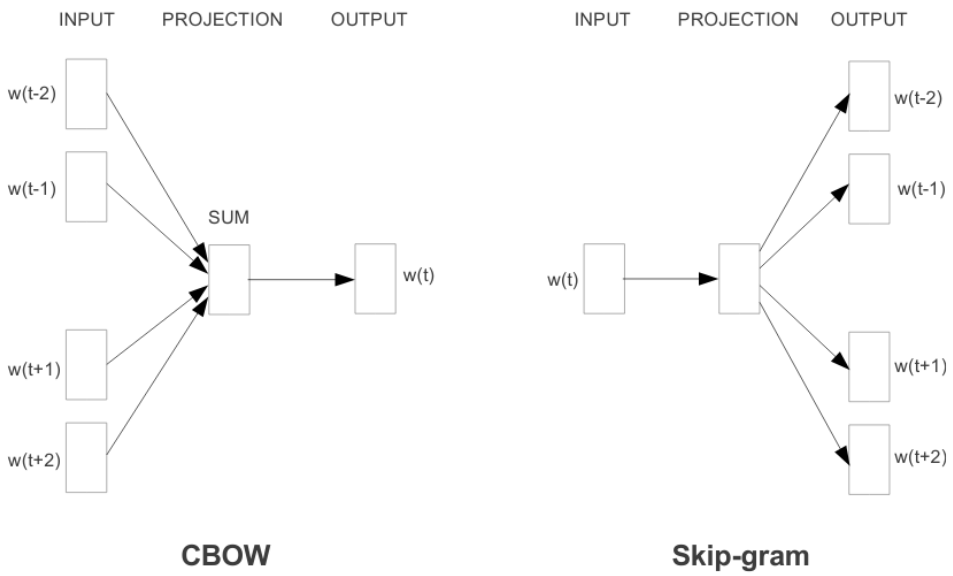

Figure 1: Word2Vec model architectures with learning objectives. Figure from Mikolov et al. [2013a]

The first released word2vec model only produces word representations for English text but it was trained with the same learning objective on corpora of other languages more recently. For my analysis, Spanish[8] and Italian [Gennaro et al., 2020] word2vec embeddings are utilised besides English[9] ones that have all been trained using Skipgram, as this strategy has proven itself more effective in evaluation tasks [Gennaro et al., 2020] than CBOW. An interesting property of the generated vectors is, that they capture linguistic relationships in vector operations: v(*Madrid*) - v(*Spain*) + v(*France*) is very close to v(*Paris*), depicting the capital-country relationship [Mikolov et al., 2013a].

To apply the general text representation of word2vec embeddings to the binary tweet classification task, the average of all word vectors of a tokenised tweet is used as input feature for a Support Vector Machine (SVM). Referring to the notation in Kulkarni and Harman [2011, chapter 17], the idea behind SVMs is to map the input features of the training examples $(\bar{x_1}, y_1)$, ..., $(\bar{x_n}, y_n)$, where $\bar{x_i} \in R^d$ and $y_i \in \{-1, 1\}$, non-linearly to a high dimensional vector space H and use a linear classifier in this new space to find a hyperplane with maximal margin to separate the transformed data $(\phi(\bar{x_i}), y_i)$,..., $(\phi(\bar{x_n}), y_n)$. New training examples are also mapped to the high dimensional space and classified depending on which side of the hyperplane

8 Github: https://github.com/dccuchile/spanish-word-embeddingsskipgram
9 https://code.google.com/archive/p/word2vec/

they lie. A separating hyperplane fulfils the following equation:

$$\bar{w} \cdot \bar{z} + b = 0 \tag{1}$$

with the vector $\bar{w} \in H$, the point $\bar{z} \in H$ and a scalar b.

$$\bar{w} \cdot \bar{z} + b > 0 \tag{2}$$

applies for all points on one side of the hyperplane, whereas

$$\bar{w} \cdot \bar{z} + b < 0 \tag{3}$$

applies for all points on the other side. Finding the optimal hyperplane to solve problems that cannot be separated linearly is an optimisation task. However, there is not always a hyperplane that separates the data points perfectly, but the objective to maximise the margin and find the most balancing separation remains.

3.2.2 BERT and mBERT

The core cross-lingual experiments of this thesis involve BERT, more specifically multilingual BERT (mBERT). In this section, the details of the state-of-the-art language model are explained, referring to the original work of Devlin et al. [2018] and the Transformer explanations in The Illustrated Transformer [Alammar, 2018]. The multilingual adaptation is discussed subsequently.

As with word2vec, the fundamental idea behind BERT embeddings is the generation of general word representations that can be leveraged with minimal effort as a feature extractor for a range of different NLP tasks. I will first look into the pre-training for learning the general-purpose language representations and then explain how the language model is used to train a classifier on task-specific labelled data. As the name BERT (**B**idirectional **E**ncoder **R**epresentations from **T**ransformers) suggests, the underlying architecture is a multi-layer bidirectional Transformer encoder. This means, it consists of stacked encoder layers, 12 in the case of BERT-base and 24 for BERT-large. Each of these encoder layers is made out of a self-attention layer and a feed-forward neural network that pass on vector representations. For the first encoder layer those are sequence embeddings but for the following blocks it is simply the output of the previous encoder layer. To facilitate a range of down-stream tasks, BERT

can handle single input sentences as well as pairs. Note that a sentence in this work does not have to be a sentence in the linguistic way but rather a contiguous text. The input embeddings E of all sequences consist of the sum of WordPiece embeddings, segment embeddings, that mark which sentence (A or B) the tokens belong to, and position embeddings that contain the tokens' absolute position in the sequence.

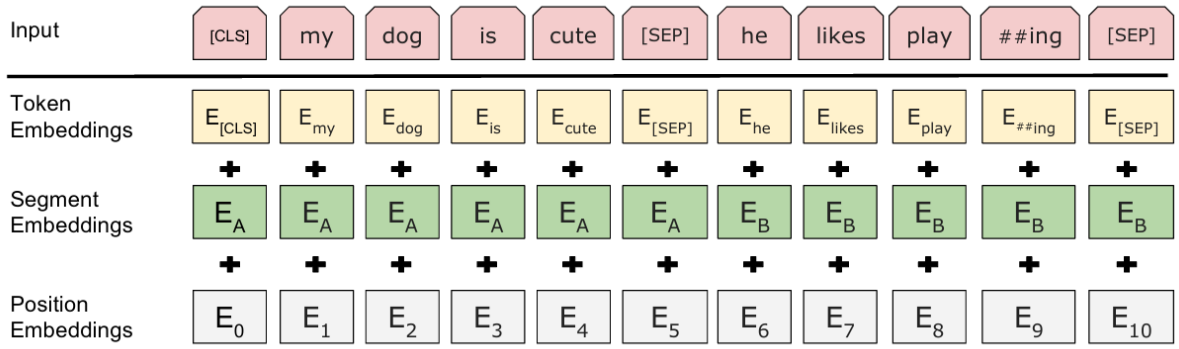

Figure 2: BERT input representation for sequence, source: Devlin et al. [2018]

WordPiece embeddings, as an approach to handle rare and unknown words, rely on a vocabulary of sub-word units, *WordPieces*, rather than full words and balance flexibility and efficiency as a middle ground between character and word delimited models [Wu et al., 2016]. The WordPiece algorithm is very similar to Byte Pair Encoding which is the equivalent tokenisation strategy for LASER, discussed in the next section. The example in Figure 2 consists of two input sentences, where the word *playing* is split into two WordPieces. Note the special tokens [SEP], to separate the sentences A and B, and [CLS] which is an additional classification token. Its final vector representation C encodes the relationship between sentence A and B during pre-training and is used by a shallow classifier during fine-tuning for a classification task, see Figure 3. The length of a sequence embedding is equal to the amount of WordPiece tokens including the special tokens. The final hidden vectors of the input tokens are denoted with $T_i \in R^H$ and in the special case of [CLS] with $C \in R^H$ where H specifies the hidden size.

With the vector representations from the Transformer encoder, the BERT model is pre-trained using self supervised learning on unlabelled text data. For that, it utilises two learning objectives. The first one is Masked Language Modelling, which describes a strategy to mask 15% of the input tokens randomly, either with a [MASK] token, a random word or the original word, and then predict the masked token based on its vector representation T. The reason to sometimes use a random or the original token for masking is to account for the missing

[MASK] token in the fine-tuning phase. To also learn relations between two separate input sentences, which is required for many downstream tasks, BERT additionally trains Next Sentence Prediction, determining whether sentence A is preceding sentence B. In half of the cases, two consecutive sentences from the training corpus are fed to the model and in the other half, two random sentences are fed. As mentioned above, the vector representation C encodes the relationship between A and B.

Both strategies combined build a strong learning objective during pre-training. The nature of the paired input provides the architectural prerequisite to easily adapt the model for task-specific fine-tuning. With task-specific input and an additional task-specific layer, BERT can be turned into different applications for a series of NLP tasks. For single sentence text classification, no second input sentence B is fed to the model but only sentence A with an empty placeholder. The binary or multi-label classification of A is realised with a linear softmax output layer on top of the pre-trained BERT that classifies the aggregated sequence representation C from the [CLS] token, see Figure 3.

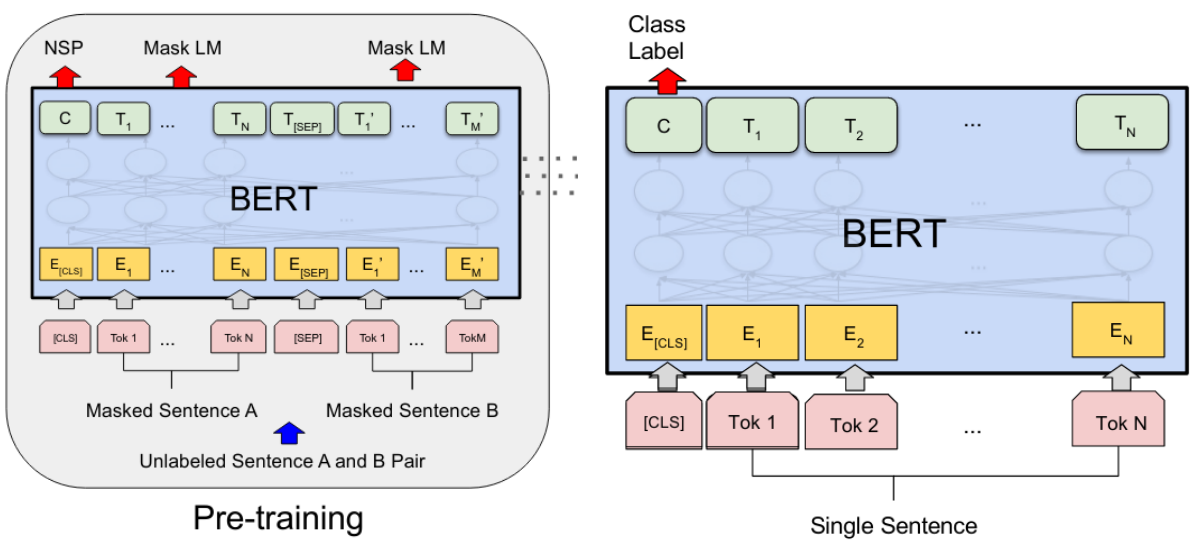

Figure 3: These figures from Devlin et al. [2018] compare pre-training of general sequence representations (*left*) and fine-tuning for single sentence classification (*right*) and show the similarity of both architectures. Note that the *left* figure is a cropped version of the original figure in Devlin et al. [2018].

More recently, two multilingual BERT models have been released that are pre-trained with the same strategy as BERT-base and BERT-large but on concatenated, not cross-lingual aligned Wikipedia dumps of 104 different languages (102 for the uncased model). To compensate for the different sizes of the Wikipedia dumps (i.e. high vs. low resource languages),

the distribution of languages is scaled by a factor smaller than one to smooth it before sampling training examples from it[10]. mBERT is trained on the obtained multilingual corpus and learns multilingual embeddings. Because WordPieces are shared across languages, the amount of overlapping subwords between source and target language positively affects the transfer ability in zero-shot scenarios across several NLP tasks [Wu and Dredze, 2019].

3.2.3 LASER Embeddings with Feed-Forward Neural Network

One of the drawbacks of mBERT is that its task-specific fine-tuning entails further tuning of the language model parameters which leads to different vector representation of the same statement in different languages depending on the source language the model is fine-tuned on. With LASER (**L**anguage-**A**gnostic **SE**ntence **R**epresentations), Artetxe and Schwenk [2019] introduce multilingual sequence embeddings that are language independent. This means they encode sentences of similar semantic content with similar representations in one shared vector space, regardless of their language (see an exemplary visualisation in Figure 4). This section describes the approach of Artetxe and Schwenk [2019] to generate these multilingual embeddings, followed by an explanation how they can be leveraged for cross-lingual text classification.

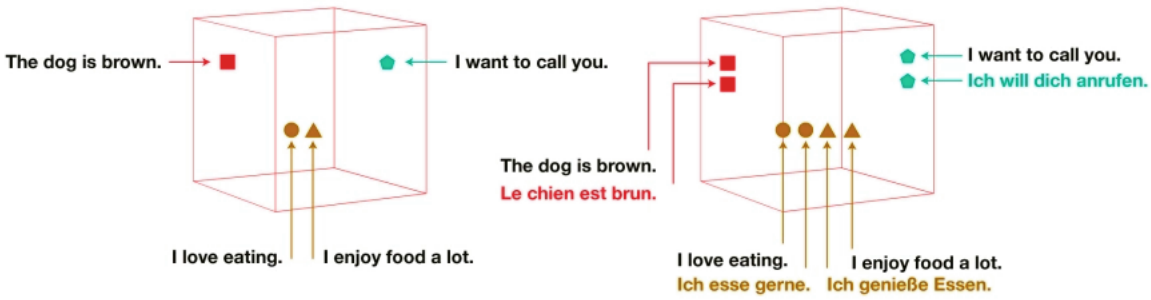

Figure 4: This figure from Schwenk [2019] shows a comparison between a regular monolingual vector space and the shared multilingual vector space of LASER embeddings. Note that the colours of the original figure have been inverted for readability reasons.

Figure 5 shows the underlying encoder-decoder architecture for pre-training, based on the work of Schwenk [2018]. The encoder is a bidirectional LSTM with five stacked layers that is

10 Github: `https://github.com/google-research/bert/`, see: multilingual.md

later used for the generation of vector representations for downstream tasks. It is trained together with a decoder LSTM on a parallel corpus of 94 languages aligned with English and Spanish.

LSTM, short for Long-Short-Term-Memory, is a special form of Recurrent Neural Networks that is introduced by Hochreiter and Schmidhuber [1997]. It manages which context information to keep and which to forget using gates, composed of a feed-forward layer, a sigmoid activation function and a pointwise multiplication with the respective layer [Jurafsky and Martin, 2019, chapter 4.9.1].

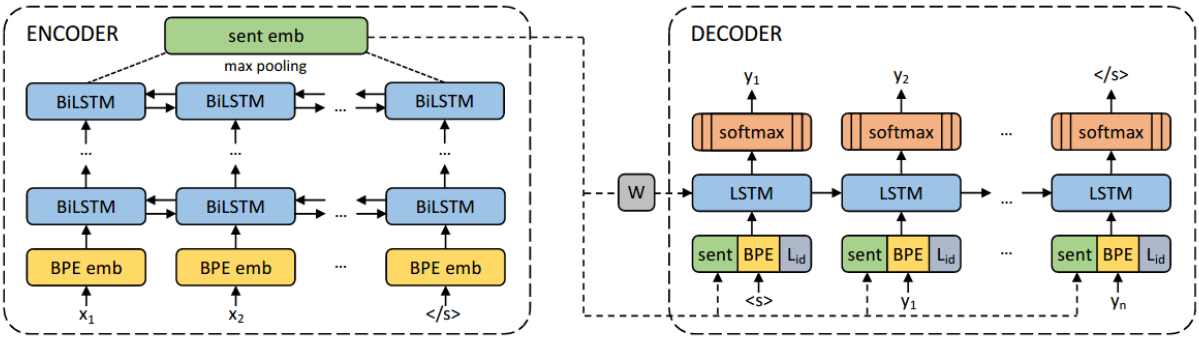

Figure 5: This figure from Artetxe and Schwenk [2019] shows the encoder-decoder architecture to learn multilingual LASER embeddings.

For LASER, an input sentence x of the training corpus is tokenised and embedded on the basis of a byte pair encoding vocabulary, before being fed as single byte pair embeddings BPE_1, ..., BPE_n to the encoder. The encoder then generates a representation for the whole sentence without knowing the source language of its input. Leaving the input language unspecified, pushes the generation of language-agnostic sentence embeddings in a multilingual vector space.

Byte pair encoding is originally a data compression technique that has been adapted by Sennrich et al. [2016] for sequence tokenisation to merge the most common characters and character sequences for generating a sub-word vocabulary. It is very similar to the WordPiece algorithm but merges sub-tokens by frequency and not likelihood of the language model [Wu et al., 2016]. By leveraging a multilingual byte pair encoding vocabulary that is shared for all languages of the training corpus [Artetxe and Schwenk, 2019], the encoder is able to create language independent sentence representations. These final sentence representations are obtained through max-pooling of the last BiLSTM layer and are further passed to the decoder, that generates translations of the encoded sequences in a specified target language

that is either English or Spanish. The target language is fed to the decoder as a language ID embedding L_{id}, that is concatenated with the whole sentence embedding, as well as the single byte pair encoding of each token. The loss of the decoder's prediction is used to update the weights of the encoder. The decision to only use two target languages for the decoder predictions is due to reasons of time and data resources but does not affect the performance in comparison to a N-way parallel corpus [Artetxe and Schwenk, 2019]. To be able to train the encoder for English and Spanish as well, two target languages are used instead of one. As with BERT embeddings, the pre-trained LASER sequence representations can be leveraged for downstream tasks like sentence classification. However, no additional fine-tuning of the language model is required. A simple feed-forward neural network can be trained as classifier on top of the pre-trained encoder part of the above described architecture. Because the LASER encodings are fixed and not tuned towards the language of the task-specific labelled data, the embeddings remain truly multilingual which is optimal for a zero-shot application. Therefore, it is even possible to train the classifier on different source languages at the same time.

4 Experiments

For evaluating a model's cross-lingual text classification performance, a comparison with similar monolingual architectures can help to establish an upper bound. The following chapter will describe different monolingual and multilingual experimental setups, including the utilised machine learning models as well as the datasets used for training, development and testing. The according results are reported in Chapter 5.

4.1 Monolingual Experiments

For the monolingual experiments, I consider Spanish and Italian only, as those are the target languages for later described cross-lingual studies. Using 10-fold cross validation with a 10-1 split to train and test on the same dataset, I compare a simple SVM classifier using word2vec embeddings with the deep learning BERT architecture described in the previous chapter. Downsampling of the majority class is applied on the respective train sets to overcome possible bias towards one label due to the uneven distribution, see Table 2. Further, 10% of the train data are split of as validation set to evaluate during training. Performance as well as standard deviation of F1 score between folds are reported.

4.2 Cross-lingual Experiments

To study the transfer ability of different text classification models, their performance when trained on tweets of one language and tested on tweets of another language is evaluated in different setups. More specifically, the transfer from English to Spanish and English to Italian as well as the transfer from Italian to Spanish and vice versa is analysed. To facilitate these experiments, two fixed test sets in Spanish and Italian are utilised, as reported in Table 2. The models are trained on a downsampled dataset like in the monolingual experiments. The following scenarios are examined:

1) **zero-shot:** neither train nor test data is translated

2) **translate-train:** train data is machine translated to test data language, training and evaluation are conducted in target language

3) **translate-test:** test data is machine translated to the training language, training and evaluation are conducted in source language

However, not every of the above described variants is evaluated with every model described in the next subsection. While experiments with and without translated data can easily be performed using a BERT architecture, word2vec embeddings for SVM are language dependent and therefore not suitable for a zero-shot scenario. LASER on the other hand, learning from multilingual sentence embeddings, does not need translation and can, after training on one language, directly be applied to data in another language not seen during training.

4.3 Bootstrap Test

To validate the reported performance differences between two cross-lingual models, a bootstrap test is performed following the algorithm described in Jurafsky and Martin [2019, chapter 4.9]. It measures with which probability the null hypothesis - that there is no statistically significant performance difference between two models - can be rejected. To determine the p-value for two models, 100 pseudo test sets $x^{*(i)}$ of the same size are created by sampling from the original test set x with replacement. For each so-called bootstrap sample the performance difference $\delta(x^{*(i)})$ between two models is computed. The final p-value results from the percentage of samples for which $\delta(x^{*(i)}) > 2\delta(x)$ with $\delta(x)$ being the performance difference between model A and B on the original test set. The intuition behind the algorithm is that model A performs better on the original test set than model B by $\delta(x)$. Because the pseudo test sets are sampled from x, one expects a very similar performance difference $\delta(x^{*(i)})$. The bootstrap tests measures how often A beats the expectation, therefore the expected value $\delta(x)$ is subtracted from each pseudo test or in fact added to $\delta(x)$.
The advantage of this bootstrap test is that it can be used for any metric, in case of this work for the F-measure. The p-values are reported in Table 6. Note that while a p-value $< .05$ suggests statistical significance, it never proves the hypothesis of statistical significance completely but strongly supports it.

4.4 Implementation and Experimental Setup

The whole implementation is done with Python3 and includes several packages and machine learning libraries described in the following subsections. For measuring performance, the F1 score is reported along precision and recall, as it represents a weighted average of these two metrics and is commonly used for classification when dealing with imbalanced class distributions.

4.4.1 Support Vector Machine

For the Support Vector Machine I use the implementation of scikit-learn[1] with a polynomial kernel to introduce non-linearity and a training duration limit of 3000 epochs.

4.4.2 BERT

All mono- and cross-lingual BERT experiments are implemented using pytorch[2] and the simpletransformers[3] library which is coupled with huggingface/transformers[4] that provides pre-trained models of Natural Language Understanding architectures like BERT. For the experiments, three different pre-trained models are used:

1) **bert-base-spanish-wwm-uncased**[5] (pre-trained by Departamento de Ciencias de la Computación Universidad de Chile) for monolingual

2) **bert-base-italian-uncased**[6] (pre-trained by MDZ Digital Library team at the Bavarian State Library) for monolingual

3) **bert-base-uncased**[7] (pre-trained by Google Reseach) for monolingual

4) **bert-base-multilingual-uncased**[8] (pre-trained by Google Research)

1 sklearn.svm.SVC
2 https://pytorch.org/
3 Simpletransformers: https://simpletransformers.ai/; Github-Repository: https://github.com/ThilinaRajapakse/simpletransformers
4 huggingface/transformers Github: https://github.com/huggingface/transformers
5 Github: https://github.com/dccuchile/beto
6 Model card: https://huggingface.co/dbmdz/bert-base-italian-uncased
7 Github: https://github.com/google-research/bert
8 Github: https://github.com/google-research/bert/blob/master/multilingual.md

All four of these models have been pre-trained on large corpora in the respective language or in the multilingual case on 100 different Wikipedia dumps. The decision to use uncased instead of case-sensitive models for my analysis is due to the nature of colloquial Twitter data that often is not correctly cased.

For the fine-tuning on preprocessed crisis tweets, a task-specific classification layer is added on top of the BERT architecture. A range of different hyperparameter combinations, that control the process of learning model weights, is evaluated to maximise the respective model's performance, as they impact small datasets like CrisisLex much stronger than bigger ones [Devlin et al., 2018]. Namely, the hyperparameters learning rate (2e-5, 3e-5, 5e-5), batchsize (16, 32), weight decay (0.01 or none) and number of epochs (2, 3, 4, 5) can profit from tuning and are therefore adjusted in different experiments. Further fixed settings are a learning rate warmup during the first 10% of steps and the use of Adam optimizer [Kingma and Ba, 2015], with an epsilon of 1e-8. These parameter values are oriented towards the recommendations of Devlin et al. [2018]. In addition, each input sequence is truncated after a maximum length of 128 WordPieces due to resource restrictions, but a previous evaluation showed that no preprocessed tweet is actually affected by this limitation. The final hyperparameter settings are reported in the appendix.

For training the BERT classifier, an additional classification token [CLS] is required in the beginning of each sequence as well as an [SEP] token in the end. During fine-tuning, the final hidden state corresponding to the [CLS] token is used as an aggregated sequence representation for classification.

As will be further discussed in the next chapter, the fold variance of monolingual cross-validation is high compared to the same splits used with the SVM. To overcome this problem, additional experiments with ensembling of models, trained on different train-dev splits using their majority vote, are conducted.

Due to the complexity of the architecture and the variety of hyperparameters, I simulate another monolingual BERT tweet classification experiment from Alam et al. [2020] to verify my implementation. The task and experimental setup are fairly similar to my binary crisis tweet classification, with the difference that another, only English, crisis dataset is used. Recreating the experiment with the same data, preprocessing steps and training procedure allows a comparison based on the reported performance and verification of my implementation, since I am able to reproduce the results with a slight improvement. This kind of comparison is not possible for the actual Spanish and Italian classification subjects because there is no literature that performs a similar analysis on the respective datasets. However, since the architecture for text classification remains the same, independent of the data language, the work of Alam

et al. [2020] is sufficient for validating my implementation.

4.4.3 LASER

To evaluate LASER embeddings, a simple feed-forward neural network as described in Artetxe and Schwenk [2019] is implemented with two hidden layers of size 10 and 8 and a Tanh activation function. During hyperparameter tuning, learning rate (0.001, 2e-5, 3e-5, 5e-5), batchsize (12, 16, 32, 64), number of epochs (2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 50, 100) and the dropout value (0.1, 0.2) are optimised. Again, the final hyperparameter settings are reported in the appendix.
For the embedding generation, the Python library laserembeddings[9] is used which enables the transformation of natural language sentences into language independent vectors utilising a pre-trained language model[10].

9 https://pypi.org/project/laserembeddings/
10 Github: https://github.com/facebookresearch/LASER

5 Results and Analysis

This chapter summarises the results of my experiments regarding monolingual and cross-lingual text classification approaches. After a short discussion of the single language models that serve as an upper bound, I will look at the core challenge of cross-lingual classification. In a further analysis, quantitative errors as well as learning behaviour of different approaches are compared.

5.1 Monolingual

As already described in the last chapter, the main difference when training monolingual models compared to cross-lingual ones is the use of the same dataset for training and evaluating by leveraging 10-fold cross-validation. The results in Table 3 report the average across all folds. However, this procedure can introduce the issue of varying fold performance, where metrics like the F1 score fluctuate significantly between different train-test splits. Interestingly, this problem occurs only within BERT experiments, although the same splits are used for both architectures by keeping a fixed random seed when applying k-fold splitting.
The intuition to stabilise the transformer's performance by creating a model ensemble, where several models are trained with different train-dev splits and then determine the final vote for the test data by majority, cannot be confirmed, the standard deviation remains high with 0.02 (0.03), see Table 3. Nevertheless, ensembling models brings a slight improvement of overall performance with up to 0.5 F1 score points which is expected, since the ensembler has seen a bigger total share of the train set.
The comparison of different ensembler sizes for the Spanish case implies that there is no infinite performance gain when increasing the number of ensembler votes. This means, there is only so much information that can be learned from different portions of the original train set. It appears that the ensembling method reaches its peak when using five models in the Spanish case, see Table 3. Therefore, only one ensembler model is trained for Italian data, due to reasons of time.
The comparison between statistical and deep learning models for monolingual crisis tweet classification shows a surprisingly strong performance of the simple SVM. With +0.9 (Spanish) and only -2.3 (Italian) F1 score points it competes with the much more complex BERT

algorithm. Both approaches appear to have the tendency to classify a tweet rather as informative than non-informative and therefore cover more actual positive tweets, which can be seen on the higher recall than precision value.
There is no indication that the dataset in one language is significantly more difficult to classify than the dataset in the other language.

Model	**F1 Score**	**Precision**	**Recall**	**F1 score σ**
SPANISH				
SVM	**86.9**	83.8	**90.3**	0.0003
BERT	85.5	83.6	87.5	0.02
BERT + ensembling (3x)	85.5	84.4	86.5	0.03
BERT + ensembling (5x)	86.0	**84.9**	87.1	0.03
BERT + ensembling (7x)	85.9	**84.9**	87.0	0.02
ITALIAN				
SVM	84.6	76.7	**94.3**	0.0006
BERT	86.5	82.8	90.5	0.02
BERT + ensembling (5x)	**86.9**	**83.1**	91.1	0.02

Table 3: Comparison of statistical and deep learning models for monolingual crisis tweet classification using 10-fold cross validation. Reported results represent average performance across folds as well as standard deviation of F1 score across folds.

Because the comparably strong SVM performance on the Spanish and Italian data questions the validity of my BERT architecture, a verification experiment is conducted, to remove doubts regarding mistakes in preprocessing, coding and experimental setup. The experiment compares the performance of my BERT classifier with a similar one described in Alam et al. [2020] and shows that my version produces proper results, that are comparable with the implementation of the literature, see Table 4. Note that the performance is reported through weighted metrics because no downsampling to account for the uneven class distribution is performed in advance.

Model	**wt. F1 score**	**wt. Precision**	**wt. Recall**	**Accuracy**
BERT Alam et al. [2020]	86.5	86.6	86.6	86.6
BERT - this work	**88.0**	**88.0**	**88.0**	**87.9**

Table 4: Performance comparison of BERT classification models using same experimental setup, parameters and dataset. Metrics: Weighted F1 score, precision and recall, accuracy.

5.2 Cross-lingual

The research question which is tackled in this thesis is the transfer ability of classification models across languages. Classifier, training language, use of translator, training scenario and test set are different aspects under which the results may be interpreted and I am going to discuss them successively. Since Spanish and Italian are linguistically closer than either of these languages with English, one could expect similar performance trends for both, however, this is not the case. In the following section I am going to point out similar and diverging trends for both languages, while taking into account the statistical significance of the results indicated by the bootstrap tests, see Table 6. A further discussion of the results follows in the next chapter.

Experimental Setup		**Zero-Shot**		**Translate-Train**		**Translate-Test**	
Model		mBERT	LASER	mBERT	SVM	mBERT	SVM
Source	**Target**						
EN	*ES*	73.9	**79.3**	77.2	**77.7**	74.9	76.0
EN^D	*ES*	73.1*	78.4*	72.2*	76.2*	74.9*	**77.1***
IT	*ES*	73.1	69.3	73.9	72.0	69.1	70.0
Source	**Target**						
EN	*IT*	62.3	66.3	69.1	68.7	68.3	68.6
EN^D	*IT*	58.0*	66.2*	68.0*	65.6*	68.4*	**70.4***
ES	*IT*	63.1	**67.0**	**69.7**	64.6	66.8	66.3

Table 5: Cross-lingual text classification performance (F1 score) across models; EN^D marks downscaled English train set, adjusted to size of respective other train set, the respective results are marked with * because they have been averaged across three different downscaled datasets.

As mBERT is evaluated in every of the three scenarios, it is a good starting point for the comparison. Table 5 shows that when trained on English tweets, mBERT's zero-shot performance can be improved by applying machine translation to either train or test data. An exception is formed by the downscaled English train set for Spanish, that does not experience a boost through translation. Including machine translation to enhance a model's transfer ability has already been proven successful in Hu et al. [2020] and also works for training on Spanish and testing on Italian tweets. However, that does not hold completely true for the learning direction Italian-Spanish, where translating the train data brings no significant

improvement ($p = .10$) and translating the test data actually worsens the results compared to zero-shot.

Model A	Model B	*P*
SVM translate-train EN - ES EN - IT	SVM translate-train IT - ES ES - IT	 $< .001$ $< .001$
translate-test EN - ES EN - IT	translate-test IT - ES ES - IT	 $< .001$ $< .001$
translate-train EN - ES IT - ES EN - IT ES - IT	translate-test EN - ES IT - ES EN - IT ES - IT	 $< .001$ $< .001$ .78 .01
SVM translate-train EN - ES IT - ES EN - IT ES - IT	mBERT translate-train EN - ES IT - ES EN - IT ES - IT	 .35 $< .001$ .35 $< .001$
translate-test EN - ES IT - ES EN - IT ES - IT	translate-test EN - ES IT - ES EN - IT ES - IT	 .15 .31 .42 $< .001$
LASER zero-shot EN - ES EN - IT	LASER zero-shot IT - ES ES - IT	 $< .001$ $< .08$
LASER zero-shot EN - ES IT - ES EN - IT ES - IT	mBERT zero-shot EN - ES IT - ES EN - IT ES - IT	 $< .001$ $< .001$ $< .001$ $< .001$
mBERT zero-shot EN - ES EN - IT	mBERT zero-shot IT - ES ES - IT	 .18 $< .001$
translate-train EN - ES EN - IT	translate-train IT - ES ES - IT	 $< .001$ .08
translate-test EN - ES EN - IT	translate-test IT - ES ES - IT	 $< .001$ .04
zero-shot EN - ES IT - ES EN - IT ES - IT	translate-train EN - ES IT - ES EN - IT ES - IT	 $< .001$ .10 $< .001$ $< .001$
translate-train EN - ES IT - ES EN - IT ES - IT	translate-test EN - ES IT - ES EN - IT ES - IT	 $< .001$ $< .001$ .15 $< .001$

Table 6: Bootstrap test between cross-lingual models.

To analyse the relation between the amount of train data and a model's performance, independent of the training language, I also conduct experiments with two downscaled English train sets, whose size is the same as the one of the Italian/Spanish one respectively. For mBERT zero-shot and translate-train, the respective F1 score drops compared to using the three times bigger original English train set. It is equal to or even lower than the F1 score on Italian-Spanish and Spanish-Italian. When translating the test tweets though, English

remains the better training language, independent of its dataset size. This last observation is also true for all SVM experiments. Rather surprising is the strong SVM performance that beats or compares with mBERT in both test languages. It reminds of the strong monolingual SVM results.
As for LASER, it significantly outperforms mBERT for zero-shot when training on English, suggesting the multilingual LASER embeddings work better for linguistically distant languages than the mBERT embeddings. The results are even stronger than the ones of the translated scenarios in the Spanish case. LASER also works better than mBERT for Spanish-Italian, however not the other way around. Considering the results for the downscaled train set, English appears to be the best source language when using LASER embeddings for classifying Spanish tweets and equally suitable ($p = .08$) as Spanish for classifying Italian tweets.
To conclude, there are a few trends that can be witnessed but interestingly, the Spanish and Italian test data have their peculiarities. Overall the classification of Italian crisis tweets seems much more difficult than the one of Spanish tweets in the cross-lingual case, see the lower F1 scores in Table 5. Considering the equally strong monolingual models, the cross-lingual results suggest a lower suitability of the utilised Italian dataset for transfer learning. For zero-shot, language-agnostic sentence embeddings emerge as strongest approach while SVM and mBERT balance each other with mostly insignificant differences when including translation. A statement about the ideal source language is rather difficult as remaining labelling varieties might create additional noise.

The learning process of language understanding models can be tracked with learning curves. They show a model's performance over time, for example by measuring the cross entropy loss on training and validation set per global step. Such graphs not only give insights into the speed of knowledge growth but also indicate when a model starts to overfit, meaning it models the training examples too perfectly while recording an increase in the validation loss. Ideally, a model should be trained as long as the performance on the validation set can be improved or the validation loss decreased. The validation set serves as an approximation of the test data and is used for hyperparameter tuning.
For this cross-lingual transfer learning study however, there is a peculiarity: The validation set, based on which the hyperparameters are selected and the best model is chosen, is split of the train set in the source language. That means it consists of training tweets that may or may not be in the same language as the test tweets and can therefore only serve as a rough performance approximation.

In Figure 6 and Figure 7 the learning curves of training and evaluation loss for mBERT trained on four different datasets are depicted. As fine-tuning a pre-trained language model is a short process, only very few iterations over the train set (epochs) are required to stabilise the validation loss.

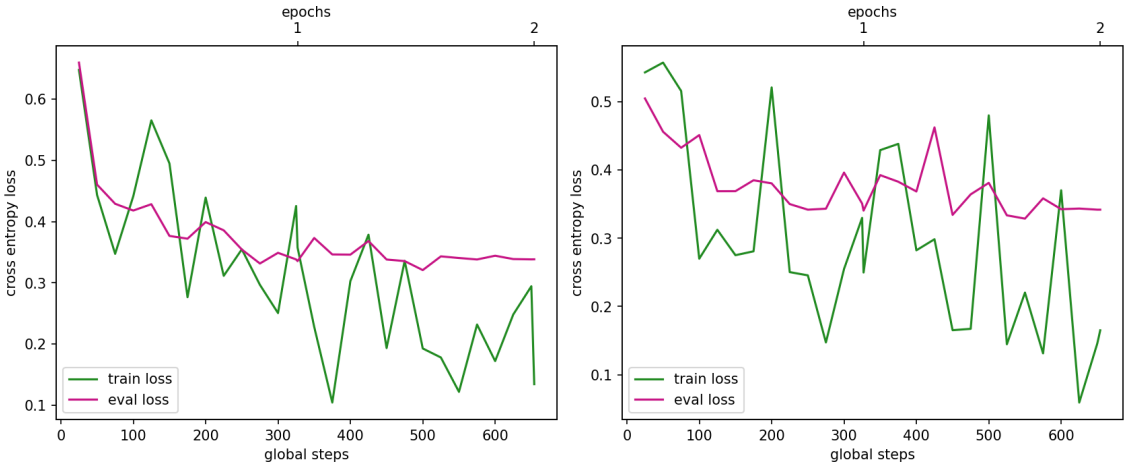

Figure 6: mBERT learning curves, *left:* English train data, *right:* English train data translated to Italian

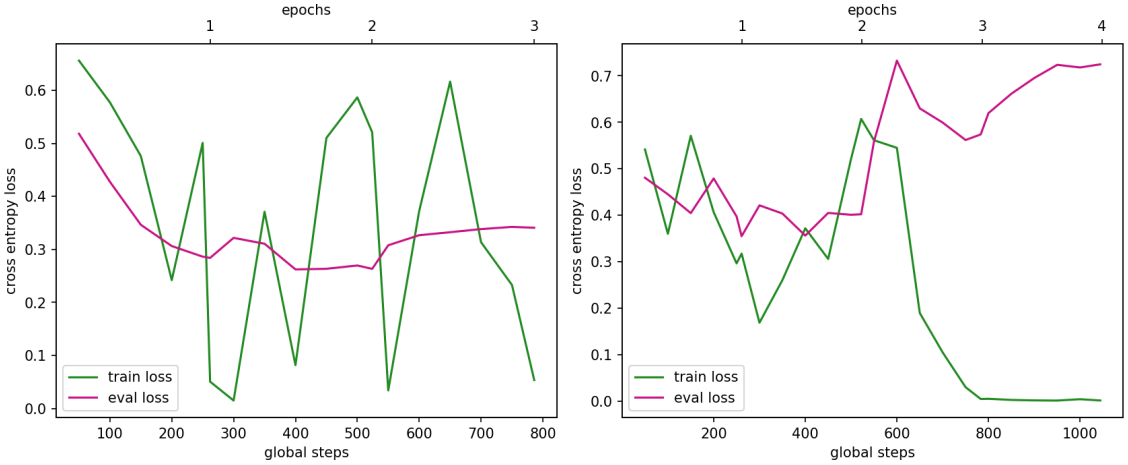

Figure 7: mBERT learning curves, *left:* Italian train data, *right:* English train data translated to Italian and downscaled to size of Spanish dataset

While the validation loss on the big English dataset decreases over the course of two epochs (Figure 6), the models trained on the smaller Italian dataset and the downscaled English set already show signs of overfitting before finishing the second epoch (Figure 7). This can be explained by their three times smaller sizes, which usually encourage the optimal modelling of the train data at the expense of the generalisation ability of the

model.

5.3 Analysis

To get more insights into the challenges of automatic classification, this section gives a quantitative overview of different tweet features that make it easier or harder to produce correct predictions. In particular, the presence or absence of URLs (substituted with an UL token) and user mentions, e.g. @denverpost, (substituted with an UQ token) is investigated, as well as the length of a tweet. Another interesting study objective would have been the use of emoticons but unfortunately, the number of tweets with emoticons in the multilingual dataset is too small for any generalisations.
Table 7 and Table 8 contain four columns that report the F1 score on all tweets with and without the above mentioned text features URL and user mention respectively. Again, the uneven class distribution per category needs to be taken into account, wherefore the weighted average of precision and recall is used for comparison.

Source	Setup	Model	URLs	No URLs	User Mentions	No User Mentions
EN	zero-shot	mBERT	**86.0**	62.3	73.2	**75.3**
EN	zero-shot	LASER	**87.3**	72.5	**80.0**	78.3
EN	translate-train	mBERT	**86.9**	68.4	77.2	77.2
EN	translate-train	SVM	**89.8**	68.3	**79.1**	74.7
EN	translate-test	mBERT	**86.0**	64.6	74.6	**75.7**
EN	translate-test	SVM	**85.2**	69.5	**77.2**	73.4
IT	zero-shot	mBERT	**80.9**	66.8	**77.6**	61.3
IT	zero-shot	LASER	**78.1**	62.6	**73.1**	60.1
IT	translate-train	mBERT	**82.6**	66.9	**77.3**	65.5
IT	translate-train	SVM	**85.3**	64.9	**76.8**	65.8
IT	translate-test	mBERT	**77.7**	61.9	**74.6**	54.9
IT	translate-test	SVM	**82.5**	62.7	**75.7**	58.9
Informative tweets			833	1071	1295	609
Non-informative tweets			257	1560	1064	753
Total counts of tweets			1090	2631	2359	1362

Table 7: Classification performance by tweet features, reported as F1 score on Spanish test data.

The first thing to note about these two tweet features is, that their presence strongly indicates informativeness on both test sets, see the total class counts at the bottom of Table 7 and Table 8. Especially the occurrence of one or more URLs seems to raise the probability for an informative tweet, suggesting the possible sharing of important crisis information.

Source	Setup	Model	URLs	No URLs	User Mentions	No User Mentions
EN	zero-shot	mBERT	**69.4**	56.8	**62.6**	62.2
EN	zero-shot	LASER	**75.0**	60.6	**77.4**	61.3
EN	translate-train	mBERT	**75.9**	64.2	**79.1**	64.6
EN	translate-train	SVM	**76.7**	63.0	**78.8**	63.6
EN	translate-test	mBERT	**73.9**	64.1	**71.1**	67.0
EN	translate-test	SVM	**76.4**	64.0	**78.8**	64.1
ES	zero-shot	mBERT	**72.5**	56.5	**70.6**	59.8
ES	zero-shot	LASER	**74.5**	62.7	**80.3**	61.5
ES	translate-train	mBERT	**76.1**	65.6	**80.9**	64.7
ES	translate-train	SVM	**76.4**	57.5	**77.5**	59.3
ES	translate-test	mBERT	**74.7**	61.5	**76.8**	62.3
ES	translate-test	SVM	**75.6**	60.3	**77.0**	61.8
Informative tweets			884	1440	778	1536
Non-informative tweets			618	3093	569	3142
Total counts of tweets			1502	4533	1357	4678

Table 8: Classification performance by tweet features, reported as F1 score on Italian test data.

More importantly though, the presence of either one of the text features increases the model's F1 score across all scenarios and architectures as well as both test sets, with the exception of two EN-ES mBERT scenarios. Furthermore, it is interesting to see that the substitution with UL and UQ tokens works for the language understanding of contextualised sentence embeddings models as well as the simple word2vec; no decrease but an increase in performance is noted for tweets with these artificial tokens.

Another aspect that suggests a correlation with a model's classification ability is the length of a tweet. The following analysis excludes very short tweets of less than four tokens, as they have been discarded during data processing. Figure 8 and Figure 9 show the performance of mBERT, LASER and SVM across different tweet lengths. The graphs indicate that text lengths is another feature that impacts the classification ability of a model. To generalise, enough tweets of one length need to be in the test set, at least approximately 100 per length. For the Spanish data that includes tweet lengths nine to 27 and for Italian data it includes tweets of five to 27 tokens. For these windows, it can be observed that the classification gets easier with increasing token count, suggesting more valuable content to interpret. However, from a certain tweet length onward, 23 tokens for Spanish and 28 tokens for Italian, there appears to be no further notable performance improvement but if anything a slight decline. These trends can be witnessed across all model architectures and both target languages. Furthermore, it is important to note that the figures do not give insights into the actual level of difficulty for very short and very long tweets due to not sufficient test examples.

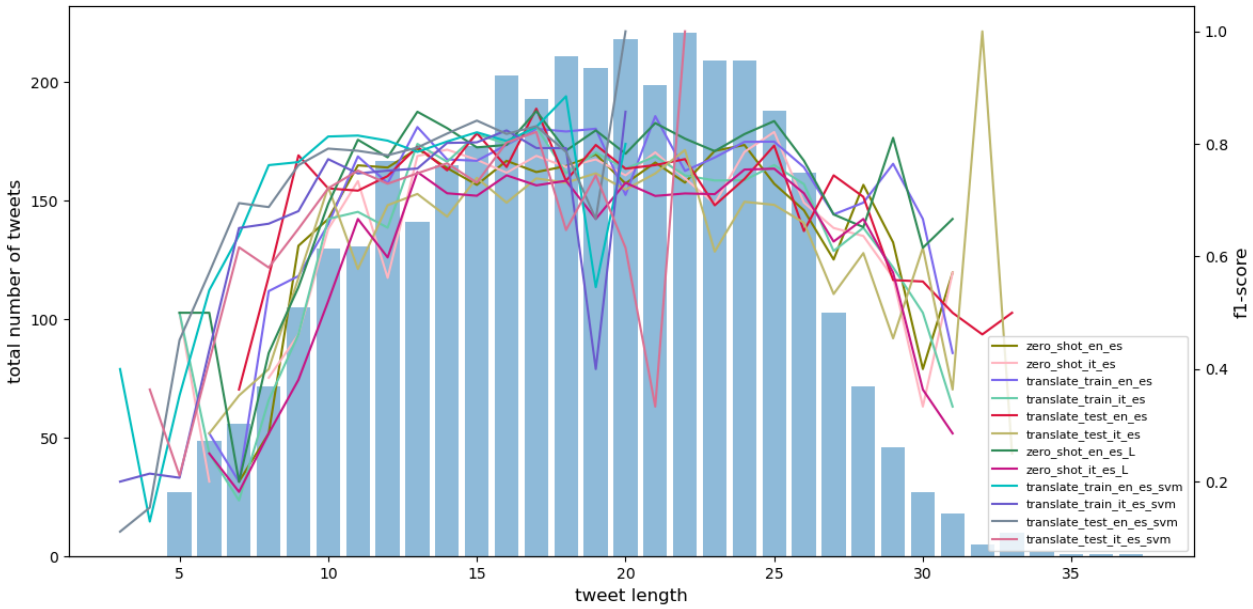

Figure 8: Classification performance by tweet length in tokens on Spanish test set.

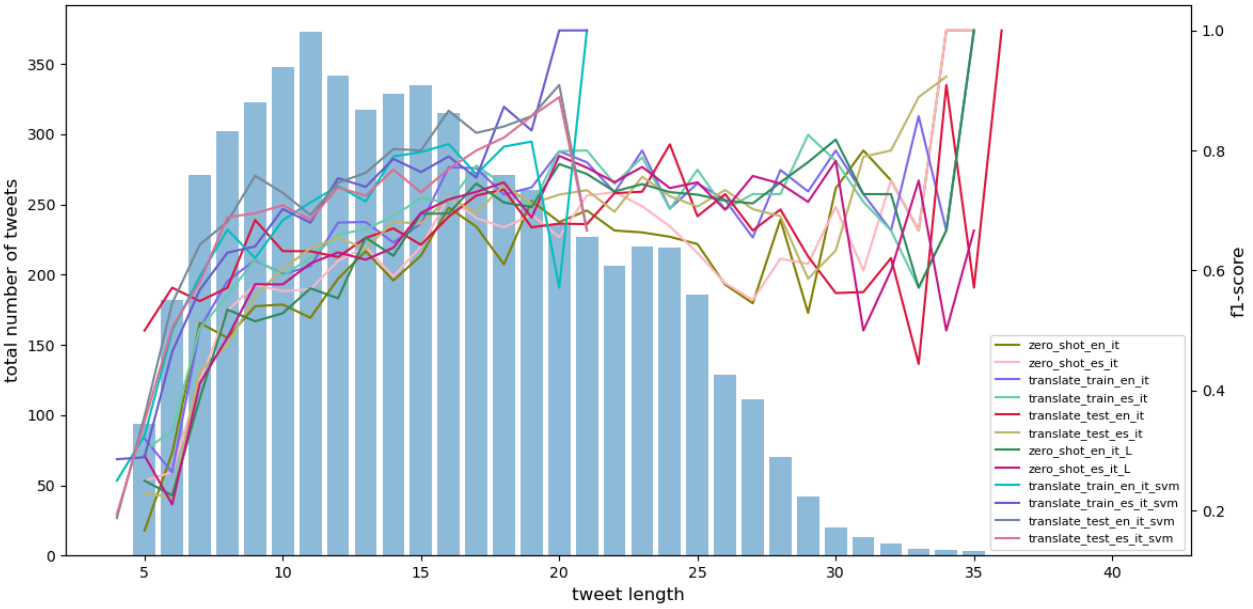

Figure 9: Classification performance by tweet length in tokens on Italian test set.

6 Discussion

In this chapter I will discuss my general procedure including data processing and experimental setup, followed by a further analysis of my findings. I will conclude with the limitations of this study, leaving room for improvement discussed in the final chapter.

6.1 Datasets and Label Fusion

For a comparative study of state-of-the-art text classification approaches tackling cross-lingual transfer learning for Twitter crisis data, I selected one big and two small datasets to simulate the high versus low resource scenario that motivates transfer learning. Though the datasets all originate from CrisisLex, a webpage for crisis-related social media data, they have been collected by different researchers introducing differing labelling systems. While previous work focuses on the simpler challenge of relatedness, this thesis tackles the more realistic question of informativeness that entails not only topic detection but more fine-grained content analysis of the tweets. For this purpose a fusion of the existing label categories into binary *informative* and *non-informative* labels is performed that adds a further disparity to the datasets of different languages. As shown in Table 3 and Table 5, there is no obvious performance gap between monolingual classification of Spanish and monolingual classification of Italian tweets. However, when looking at the cross-lingual setup, the different levels of difficulty of transfer learning for both languages become clear, indicating a more challenging transfer from English and Spanish to Italian and also partially Italian to Spanish. When recalling the label fusion in Chapter 3, it is notable that the damage label of the Italian T4 that constitutes the Italian informative category only groups tweets assessing damages to infrastructure and victims. It does not include tweets about recommendations for action, caution and general advice as well as information about a disaster itself. Both other datasets label all this information as informative. Though the final multilingual dataset also comprises Spanish and Italian tweets of T26, the number of Italian informative tweets of T26 is smaller compared to the number of Spanish informative tweets of T26. That means, that the final Spanish and English parts of the multilingual dataset have much closer informative categories than Italian and are therefore more suitable for transfer. There is no other meaningful way of merging the

categories regarding informativeness on the basis of the available labelling, but collecting a new multilingual dataset with the same labels for each language could be subject of future work.
For the data preprocessing I follow existing work and architecture-specific recommendations. Considering a real-world application of crisis tweet classification, a brief text preprocessing procedure is preferential over a complex one due to reasons of time and the immediate call for action on the basis of real-time information. Therefore, I decide to replace URLs with an UL token, omitting the possible underlying information. However, in some cases like T26 the human annotators were supposed to follow links in order to obtain additional information and differentiate between possible advertisement spam and informing tweets. It is an information for time trade-off that could be tackled with a further analysis of the URL string itself.

6.2 Experimental Setup

For comparing the different architectures, three common setups for cross-lingual transfer learning are evaluated: no translation, translation of train data to target language and translation of test data to source language. When trying to optimise the performance of machine classification, usually elaborate settings with time-consuming methods are compared. Whether it is the additional analysis of linked websites like mentioned above or the use of state-of-the-art machine translation APIs to ease the transfer learning, these add-ons can help to increase metrics but they also increase expenditure. Therefore, the real life feasibility of a model and its setup is an important factor worth considering, when working on a tool for a crisis-application for humanitarian help like a disaster tweet classifier. Omitting the use of any machine translation is one possible design decision, that is even competitive with translation scenarios when using language-agnostic text encodings like LASER. However, for languages not represented through LASER, translation for cross-lingual transfer learning is still an important component. There is no obvious answer to the question of whether to translate training tweets or tweets used for inference because both have their advantages and disadvantages: While translating train data in advance saves time during inference, it causes the need for many different classifiers each specialised for one language because the target language of crisis tweets is not known beforehand. On the other hand, translating the tweets during inference requires access to on-demand machine translation applications that are able

to process vast amounts of text in brief time.
Apart from the feasibility when used in the field, zero-shot setups have a further benefit, as they give some indication of a model's ability to abstract away from languages and understand the language-agnostic meaning of a text. The use of machine translation on the other hand, always introduces the quality factor of a translation algorithm, so the comparison becomes a comparison of machine translation algorithms rather than a comparison of text classification architectures.

6.3 Comparison of Models

As already analysed in Chapter 5.2, the different architectures, setups and source and target languages pose manyfold aspects that affect a model's performance, making it difficult to extract clear trends. In fact, there remain some open questions such as the fact that the mBERT zero-shot transfer from Italian to Spanish is more successful than the mBERT transfer from Spanish to Italian. The issue of a slight label discrepancy is already mentioned in Section 6.1 but does not explain why LASER is not affected.
Furthermore, a few cases occur, where a model's performance without translation is better than with translation, for example when trained on Italian and tested on Spanish, questioning the importance of machine translation quality in the whole setup.
Another surprising turnout is the strong SVM performance in monolingual and cross-lingual experiments. The use of contextualised text representations like BERT and LASER raises the expectation of strong performance improvements compared to the simple bag-of-words word2vec embeddings used by SVM. Instead, SVM competes with mBERT for the translated scenarios which are in essence monolingual, too. A possible reason is the nature of tweets: short, colloquial and often times phrases and no full sentences. BERT and LASER embeddings usually excel in incorporating the meaning of whole sentences and longer continuous sequences, where context dependencies like coreference resolution need to be solved, which is not primarily the case for tweets. However, the true zero-shot transfer is not tested with SVM, since word2vec language models are trained on one language only. In the future, it would be interesting to investigate whether multilingual word embeddings like MUSE [Conneau et al., 2017] can compete with multilingual sentence representations or whether the language transfer requires contextualised sequence encodings like LASER or BERT.
Last but not least, there are a few cases where a model trained on the downscaled English

dataset outperforms a model trained on the large English dataset, which happens primarily in the translate-test scenario. Since the F1 score for the downscaled train set is computed through the average on three different subsets of the English train set, a coincidental extraction of particularly meaningful tweets should be eliminated. It appears, that this trend does not occur consistently through all scenarios which further rejects the hypothesis of randomly meaningful subsets and leaves an open question.

6.4 Limitations

In order to determine future research directions, it is important to consider the limitations of one's work. This thesis conducts experiments regarding the transfer ability of cross-lingual model setups including available resources, not a direct comparison of different embedding techniques. To actually compare them, each language model for embedding generation would need to be trained on the same corpus and evaluated in the same setting, using the same classifier architecture, not a SVM on the one hand and a feed-forward neural network on the other hand. Nevertheless, this thesis approximates a direct comparison with fairly similar architectures which actually generates findings more relevant for applications in the field.

7 Conclusion

This final chapter summarises my comparative study and findings and presents further research questions. It concludes the analysis and describes how the tool of crisis tweet classification can be adopted for real world applications for humanitarian help.

7.1 Summary

This thesis aims at comparing different approaches, based on word and sentence level representations, for cross-lingual text classification of tweets during a crisis. In order to investigate the transfer ability of different models, I consolidate three publicly available tweet datasets of different languages, fusing their categories to binary informativeness labels. Each model is then trained on data of one language and evaluated in different scenarios on data of another language, with Spanish and Italian being the two target languages of interest. The results indicate that an increased size of the train set positively influences a model's performance and that the transfer direction plays a role, too. For the two scenarios including machine translation, additional variance is introduced through the quality of the translation application, which can fluctuate depending on the translation direction. However, neither one of the translation scenarios is consistently better than the other, suggesting an individual choice when used in a real world application, based on the available translation resources.
LASER embeddings emerge as the most successful approach when dealing with zero-shot scenarios. If translation is included, bag-of-words embeddings compete with contextualised BERT embeddings, questioning the use of complex architectures like mBERT for tweet classification. Tweet features like a higher token count, the use of user mentions and URLs positively affect a model's performance.
All in all, the use of language-agnostic sentence embeddings seems promising because of its feasibility in the field and its ability to handle code switching within a sequence, a phenomenon often present in tweets. It also brings up the question of the comparability of multilingual word embeddings.
Furthermore, the experiments show that the equality of source and target task ($T_S = T_T$), given through the same classification task on equal label categories with equal label meaning in source and target data, is crucial for cross-domain (cross-lingual) transfer learning. Only

with this prerequisite, the importance of linguistically close source and target languages versus big training resources can be properly compared.

This work evaluates different transfer learning approaches within one experimental scope using a single multilingual dataset. Thereby, it overcomes the issue of separate but not comparable progress on the task of cross-lingual crisis tweet classification due to different datasets and labelling systems.

7.2 Outlook

Transfer learning for cross-lingual text classification is an NLP task that is not only gaining importance in the field of Crisis Informatics. Many groundbreaking achievements in machine learning are developed on the basis of huge standard language corpora, neglecting low resource languages for which labelled data is sparse. Working on multi-purpose representations that can be used in cross-lingual settings opens the door for zero- and few-shot applications where only little to none labelled data in the target language is required. The knowledge transfer from source to target data can be optimised using suitable language pairs. As existing research implies, choosing the optimal source language is a challenge of its own that could be tackled by looking at criteria like subword overlap of the respective vocabularies [Wu and Dredze, 2019]. Enriching the source data with labelled target examples in a few-shot scenario could further help to facilitate the knowledge transfer.
As the quantitative findings of this thesis suggest, colloquial tweet data does not necessarily profit from rich contextualised sequence encodings. To get a better understanding whether sentence embeddings actually incorporate the content essence of tweets compared to context-less word embeddings, a manual qualitative analysis of the models' predictions could be interesting. It could reveal for example whether sentence embeddings are able to encode crisis-unrelated tweets that use trending crisis hashtags for popularity more nuanced than simple bag-of-words embeddings like word2vec. A more fine-grained multi-label classification as well as the component of crisis type are further research subjects that have not been worked on yet due to the lack of a big enough multilingual dataset.
This study has shown the feasibility of leveraging cross-lingual transfer learning to classify crisis tweets regarding their informativeness without any labelled target language examples. The presented strategies can also be applied to content of other social media networks like

Facebook for example. While there is room to improve classification performance with using large enough train sets from suitable source languages, the existing classification approaches, especially when not using translation, are simple enough to be applied in the field. Coupled with a Twitter streaming application with possible preliminary keyword filtering, the presented tweet classifiers can be utilised to extract valuable crisis information from affected individuals and first responders out of the mass of constantly posted content. This way, responsible authorities like intelligence services, governments and even humanitarian aid organisations can gain real-time insights and assessments of the situation during a crisis.

Bibliography

Firoj Alam, Hassan Sajjad, Muhammad Imran, and Ferda Ofli. Standardizing and benchmarking crisis-related social media datasets for humanitarian information processing. *arXiv preprint arXiv:2004.06774*, 2020.

Jay Alammar. The illustrated transformer, 2018. URL http://jalammar.github.io/illustrated-transformer/. Accessed: 2020-08-29.

Mikel Artetxe and Holger Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610, 2019.

Jens A. de Bruijn, Hans de Moel, Albrecht H. Weerts, Marleen C. de Ruiter, Erkan Basar, Dirk Eilander, and Jeroen C. J. H. Aerts. Improving the classification of flood tweets with contextual hydrological information in a multimodal neural network. *Computers and Geosciences*, 140:104485, 2020.

Alfredo Cobo, Denis Parra, and Jaime Navón. Identifying relevant messages in a twitter-based citizen channel for natural disaster situations. In *Proceedings of the 24th International Conference on World Wide Web*, page 1189–1194, 2015.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. *CoRR*, abs/1710.04087, 2017. URL http://arxiv.org/abs/1710.04087.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, E. Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 31–38, 2019.

Stefano Cresci, Maurizio Tesconi, Andrea Cimino, and Felice Dell'Orletta. A linguistically-driven approach to cross-event damage assessment of natural disasters from social media messages. In *Proceedings of the 24th International Conference on World Wide Web*, page 1195–1200, 2015.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Giovanni Di Gennaro, Amedeo Buonanno, Antonio Di Girolamo, Armando Ospedale, Francesco A. N. Palmieri, and Gianfranco Fedele. An analysis of word2vec for the italian language. *arXiv preprint arXiv:2001.09332*, 2020.

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

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *arXiv preprint arXiv:2003.11080*, 2020.

Muhammad Imran, Prasenjit Mitra, and Jaideep Srivastava. Cross-language domain adaptation for classifying crisis-related short messages. *arXiv preprint arXiv:1602.05388*, 2016.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782, 2019.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing (3rd Edition Draft)*. 2019. Unpublished Draft.

Prashant Khare, Grégoire Burel, Diana Maynard, and Harith Alani. Cross-lingual classification of crisis data. In *International Semantic Web Conference*, pages 617–633, 2018.

Prashant Khare, Grégoire Burel, and Harith Alani. Relevancy identification across languages and crisis types. *IEEE Intelligent Systems*, 34:19–28, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Sanjeev Kulkarni and Gilbert Harman. *An Elementary Introduction to Statistical Learning Theory*. John Wiley & Sons, Incorporated, 2011.

Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7059–7069, 2019.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. Choosing transfer languages for cross-lingual learning. *arXiv preprint arXiv:1905.12688*, 2019.

Valerio Lorini, Carlos Castillo, Francesco Dottori, Milan Kalas, Domenico Nappo, and Peter Salamon. Integrating social media into a pan-european flood awareness system: A multilingual approach. *arXiv preprint arXiv:1904.10876*, 2019.

Marco Lui and Timothy Baldwin. Accurate language identification of twitter messages. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)*, pages 17–25, 2014.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, page 3111–3119, 2013b.

Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the 8th International AAAI Conference on Weblogs and Social Media (ICWSM'14)*, 2014.

Alexandra Olteanu, Sarah Vieweg, and Carlos Castillo. What to expect when the unexpected happens: Social media communications across crises. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work and Social Computing*, page 994–1009, 2015.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631, 2019.

Holger Schwenk. Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, 2018.

Holger Schwenk. Zero-shot transfer across 93 languages: Open-sourcing enhanced laser library. URL: https://engineering.fb.com/ai-research/laser-multilingual-sentence-embeddings/, 2019. Accessed: 2020-08-30.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, 2016.

Johnny Torres, Carmen Vaca. Cross-lingual perspectives about crisis-related conversations on twitter. In *Companion Proceedings of The 2019 World Wide Web Conference*, page 255–261, 2019.

Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, 2019.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

A Appendix

For the sake of completeness and to allow for recreating the experiments, the appendix reports the fine-tuning hyperparameters for all neural network experiments.

A.1 Monolingual BERT Fine-Tuning

The Spanish model used is: **bert-base-spanish-wwm-uncased**, which was pre-trained by the Departamento de Ciencias de la Computación Universidad de Chile.

Parameter	**Setting**
Epochs	4
Batch size (train & eval)	16
Learning rate	$2 \cdot 10^5$
Weight decay	0.01
Adam epsilon	$1 \cdot 10^{-8}$
Maximum sequence length	128
Warm-up ratio	0.1

Table 9: Fine-tuning hyperparameters for monolingual classification using a pre-trained Spanish BERT model.

The Italian model used is: **bert-base-italian-uncased**, which was pre-trained by the MDZ Digital Library team at the Bavarian State Library.

Parameter	**Setting**
Epochs	4
Batch size (train & eval)	32
Learning rate	$5 \cdot 10^5$
Weight decay	0.01
Adam epsilon	$1 \cdot 10^{-8}$
Maximum sequence length	128
Warm-up ratio	0.1

Table 10: Fine-tuning hyperparameters for monolingual classification using a pre-trained Italian BERT model.

A.2 Cross-lingual BERT Fine-Tuning

For all cross-lingual BERT experiments, the pre-trained **bert-base-multilingual-uncased** is used, which is provied by Google Research. In total, fine-tuning is conducted on seven datasets and the trained models are evaluated in 12 test settings.

English Train Set

Fine-tuning on the original English train set is conducted three times: on the English tweets, on the English tweets translated to Spanish and on the English tweets translated to Italian.

Parameter	**Setting**		
	EN	**EN-ES**	**EN-IT**
Epochs	2	2	2
Batch size (train & eval)	32	16	32
Learning rate	$2 \cdot 10^5$	$5 \cdot 10^5$	$5 \cdot 10^5$
Weight decay	0.01	0.01	0.01
Adam epsilon	$1 \cdot 10^{-8}$	$1 \cdot 10^{-8}$	$1 \cdot 10^{-8}$
Maximum sequence length	128	128	128
Warm-up ratio	0.1	0.1	0.1

Table 11: Hyperparameters for cross-lingual classification using mBERT fine-tuned on English dataset.

Spanish Train Set

Fine-tuning on the original Spanish train set is conducted two times: on the Spanish tweets and on the Spanish tweets translated to Italian.

Parameter	**Setting**	
	ES	**ES-IT**
Epochs	4	4
Batch size (train & eval)	32	32
Learning rate	$2 \cdot 10^5$	$5 \cdot 10^5$
Weight decay	0.01	0.01
Adam epsilon	$1 \cdot 10^{-8}$	$1 \cdot 10^{-8}$
Maximum sequence length	128	128
Warm-up ratio	0.1	0.1

Table 12: Hyperparameters for cross-lingual classification using mBERT fine-tuned on Spanish dataset.

Italian Train Set

Fine-tuning on the original Italian train set is conducted two times: on the Italian tweets and on the Italian tweets translated to Spanish.

Parameter	**Setting**	
	IT	**IT-ES**
Epochs	3	4
Batch size (train & eval)	16	16
Learning rate	$2 \cdot 10^5$	$3 \cdot 10^5$
Weight decay	0.01	0.01
Adam epsilon	$1 \cdot 10^{-8}$	$1 \cdot 10^{-8}$
Maximum sequence length	128	128
Warm-up ratio	0.1	0.1

Table 13: Hyperparameters for cross-lingual classification using mBERT fine-tuned on Italian dataset.

A.3 LASER Fine-Tuning

For LASER, the task-specific fine-tuning hyperparameters are as follows:

Parameter	**Setting**		
	EN	**ES**	**IT**
Epochs	15	30	20
Batch size (train & eval)	16	16	16
Learning rate	0.001	0.001	0.001
Dropout	0.1	0.2	0.2
Weight decay	0	0	0

Table 14: Hyperparameters for cross-lingual classification using LASER.