Zero-shot Cross-Lingual Counterfactual Detection via Automatic Extraction and Prediction of Clue Phrases

Asahi Ushio[1]
[1]Cardiff University
UshioA@cardiff.ac.uk

Danushka Bollegala [2,3]
[2]Amazon, [3]University of Liverpool
danubol@amazon.com

Abstract

Counterfactual statements describe events that did not or cannot take place unless some conditions are satisfied. Existing counterfactual detection (CFD) methods assume the availability of manually labelled statements for each language they consider, limiting the broad applicability of CFD. In this paper, we consider the problem of zero-shot cross-lingual transfer learning for CFD. Specifically, we propose a novel loss function based on the clue phrase prediction for generalising a CFD model trained on a source language to multiple target languages, without requiring *any* human-labelled data. We obtain clue phrases that express various language-specific lexical indicators of counterfactuality in the target language in an unsupervised manner using a neural alignment model. We evaluate our method on the Amazon Multilingual Counterfactual Dataset (AMCD) for English, German, and Japanese languages in the zero-shot cross-lingual transfer setup where no manual annotations are used for the target language during training. The best CFD model fine-tuned on XLM-R improves the macro F1 score by 25% for German and 20% for Japanese target languages compared to a model that is trained only using English source language data.

1 Introduction

A counterfactual statement describes an event that may not, did not, or cannot take place, and the subsequent consequence(s) or alternative(s) did not take place (Milmed, 1957). Counterfactual statements can take the form – *If p was true, then q would be true* (i.e. assertions whose antecedent (p) and consequent (q) are known or assumed to be false). Counterfactual detection (CFD) is an important task in NLP, which has found broad applications such as customer review analysis in e-commerce (O'Neill et al., 2021), social media analysis (Son et al., 2017) and automatic psychological assessment (Janocko et al., 2016). To further explain the CFD task, consider the following counterfactual statement extracted from a product review: *I wish the trouser had ruching so that it could fit me well.* This is considered a counterfactual statement because it has the subjunctive mood *wished* and the author of the review wishes that the trouser had ruching, whereas it does not have in reality. In this particular example, *trouser had ruching* is the antecedent and *it could fit me well* is the consequent. Ideally, for a user who is searching for *trousers with ruching* we should *not* display this particular trouser because it does not have rouching. By accurately detecting counterfactual statements, we can prevent such irrelevant search results.

Almost all prior work on CFD has been limited to the English language (Yang et al., 2020; Son et al., 2017; Ding et al., 2020; Fajcik et al., 2020; Lu et al., 2020; Ojha et al., 2020; Yabloko, 2020) with the notable exception of O'Neill et al. (2021), who looked at German and Japanese counterfactuals in addition to English. However, *all* existing work on CFD require manually labelled language-specific counterfactual statements for the target language of choice. Extending CFD to different target languages has been hindered so far by two main challenges. First, manual annotation of counterfactuality is a time consuming and a costly task, which requires professional linguists as shown by O'Neill et al. (2021). Moreover, such expert annotators might not be available for all languages we would like to perform CFD. Second, counterfactual clues such as *wished, would have* (in English) or *fehlt, wenn es* (in German) etc. are highly language-specific, which makes it difficult to transfer a model trained on a source language to a different target language without neither labelled counterfactual examples nor clue phrase lists.

To address the above-mentioned challenges, we propose a zero-shot cross-lingual transfer learning method for CFD that learns a CFD model for a tar-

Proceedings of the The 2nd Workshop on Multi-lingual Representation Learning (MRL), pages 28 - 37
December 8, 2022 ©2022 Association for Computational Linguistics

get language without using any labelled data for that target language. Our proposed method consists of two steps: (a) **automatic clue phrase extraction** for the target language and (b) learning a CFD classifier for the target language by **predicting the clue phrases in the text**. We use a neural alignment model (Dou and Neubig, 2021) to align machine-translated source language counterfactual sentences to find clue phrases for the target language. We then use those automatically extracted target language clue phrases to induce sequential labels for the sentences in the target language to train a CFD model. For this purpose, we propose a novel training objective that consists of a main task (i.e. predicting whether a given sentence contains a counterfactual statement or not) and an auxiliary task (i.e. predicting whether a given token in a sentence is a clue phrase or not). To the best of our knowledge, we are the first to propose a transfer learning method for cross-lingual CFD, let alone in a zero-shot setting that does not require neither counterfactual clues nor labelled training instances for the target language.

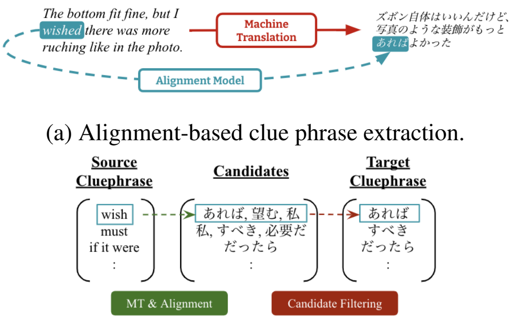
Using the Amazon Multilingual Counterfactual Detection dataset (AMCD) (O'Neill et al., 2021), we evaluate the proposed method for its ability to perform cross-lingual zero-shot transfer. Specifically, we use token-embeddings obtained from XLM-R (Conneau et al., 2019) and mBERT[1] to train CFD models for German and Japanese target languages using counterfact labelled sentences for English source language and automatically extracted clue phrases for each target language. In particular, no human counterfact annotations for the target language are used during training. Our proposed method establishes a new state-of-the-art for zero-shot cross-lingual transfer with an improvement of 25% in macro-averaged F1 score for German and that of 20% for Japanese. The Source code implementation for the proposed method will be publicly released upon paper acceptance.

2 Related Work

For training and evaluating CFD methods, a dataset was annotated in the SemEval-2020 Task 5 (Yang et al., 2020) covering two subtasks. The first subtask is to classify a given sentence as to whether it expresses a counterfactual statement or not, whereas in the second subtask the participating teams must extract the antecedent and consequent from a given counterfactual statement. Our goal in this paper is close to the first subtask, which can be modelled as a sentence-level binary classification problem. Most of the high performing methods (Ding et al., 2020; Fajcik et al., 2020; Lu et al., 2020; Ojha et al., 2020; Yabloko, 2020) submitted to SemEval-2020 Task 5 use state-of-the-art pretrained language models (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2019; Yang et al., 2019) to represent sentences. Traditional machine learning methods, such as support vector machines and random forests were also used but with less success (Ojha et al., 2020). However, none of these previously proposed methods consider cross-lingual nor zero-shot CFD settings. To achieve the best prediction quality, ensemble strategies are employed. The top performing systems use an ensemble of transformers (Ding et al., 2020; Fajcik et al., 2020; Lu et al., 2020), while others include Convolutional Neural Networks (CNNs) with Global Vectors (Pennington et al., 2014) embeddings (Ojha et al., 2020). Various structures are used on top of transformers. For example, Lu et al. (2020); Ojha et al. (2020) use a CNN as the top layer, while Bai and Zhou (2020) use a Bi-GRUs and Bi-LSTMs. Some other proposed methods use additional modules, such as constituency and dependency parsers in the lower layers of the architectures (Yabloko, 2020).

O'Neill et al. (2021) created the AMCD counterfactual dataset by annotating sentences selected from Amazon product reviews. Unlike the SemEval dataset, which covers only English counterfactuals, AMCD covers Japanese and German counterfactuals in addition to English. AMCD is the only publicly available multilingual CFD dataset. Therefore, we use AMCD to evaluate the cross-lingual zero-shot CFD models we propose in this paper. O'Neill et al. (2021) trained CFD models using different approaches such as bag-of-words representations of sentences as well as by fine-tuning pre-trained masked language models for each language separately. They also considered a cross-lingual zero-shot setting where they first machine translated the source (English) dataset into each of the target languages (German and Japanese), and train CFD models for those target languages using the translated training instances. However, the performance of this approach was significantly worse than that of the in-

[1] https://github.com/google-research/bert/blob/master/multilingual.md


The bottom fit fine, but I wished there was more ruching like in the photo.
Machine Translation
ズボン自体はいいんだけど、写真のような装飾がもっとあればよかった
Alignment Model

(a) Alignment-based clue phrase extraction.

(b) End-to-end pipeline for extracting clue phrases.

Figure 1: An example of extracting clue phrase candidates for Japanese target language from a pair of sentences obtained by machine translating an English source language sentence to Japanese. The alignment model aligns the English clue phrase *wished* with the Japanese term あれば(*areba*), which is then extracted as a candidate Japanese clue phrase.

language baselines, which lead to their conclusion "*simply applying MT on test data is not an alternative to annotating counterfactual datasets from scratch for a novel target language.*" This highlights the difficulty of the cross-lingual zero-shot transfer problem setting for CFD, which we consider in this paper.

One approach to learn accurate multilingual representations with less supervision in downstream tasks is the few-shot or zero-shot cross-lingual transfer learning. Here, the goal is to transfer a model trained in the source language into the target language with minimal loss in performance. Few-shot transfer learning assumes the availability of a small number of labelled instances in the target language, while the zero-shot setting assumes none. Recently, Pfeiffer et al. (2020) proposed MAD-X, a zero-shot and few-shot language model transfer framework based on the adaptor framework (Houlsby et al., 2019). Moreover, XTREME (Hu et al., 2020), a multilingual benchmark containing many tasks, reported the translate-train performance, where a model is trained on a machine-translated version of the source language dataset into the target languages as a baseline for zero-shot transfer learning. However, to the best of our knowledge, ours is the first-ever model proposed for cross-lingual zero-shot transfer for CFD.

3 Cross-lingual Zero-shot CFD

Let us denote a sentence $x = w_1, w_2, \ldots, w_{|x|}$ consisting of a sequence of $|x|$ tokens w_j. CFD is considered as a binary classification task in this paper, where the goal is to predict whether a sentence x contains a counterfactual statement ($y(x) = 1$) or otherwise ($y(x) = 0$), indicated by the binary label $y(x)$. In the cross-lingual zero-shot CFD setting, we consider the problem of transferring a CFD model trained on a source language s to a different target language t. For this purpose we assume the availability of a counterfactual-labelled dataset, $\mathcal{D}_s = \{(x_{s,i}, y(x_{s,i}))\}_{i=1}^{|\mathcal{D}_s|}$ for the source language and an unlabelled dataset, $\mathcal{D}_t = \{x_{t,i}\}_{i=1}^{|\mathcal{D}_t|}$, for the target language. Here, we use the notation $x_{s,i}$ to indicate the i-th sentence in the source (for the target $x_{t,i}$) language dataset and its associated counterfactual label $y(x_{s,i})$. The source language is assumed to be a language for which it is relatively easier to create a large annotated dataset because it is easier to recruit annotators than for the target language. Following prior work on cross-lingual transfer (Hu et al., 2020; O'Neill et al., 2021), we use a machine translation (MT) system to translate the sentences in $\mathcal{D}_s$ into the target language with the labels unchanged to create a machine-translated version of $\mathcal{D}_s$, denoted by $\mathcal{D}_{mt}$.

Counterfactual statements are rare in natural language sentences and Son et al. (2017) report that only 1-2% of sentences contain counterfactual statements in a random collection of sentences. Therefore, randomly selecting sentences for annotation purposes results in a waste of annotation resources such as annotator time and cost, and will only result in an imbalanced and low-coverage datasets. To address this issue, prior work (Son et al., 2017; O'Neill et al., 2021; Yang et al., 2020) creating annotated datasets for counterfactuality has used language-specific *clue phrases* that indicate various expressions frequently used to indicate the presence of a counterfactual to filter candidate sentences for annotation. We use such counterfactual clue phrases as auxiliary training data for cross-lingual transfer. Specifically, we require that a CDF model can not only (a) predict whether a given sentence x is a counterfactual or not (*main task*), but also be able to (b) predict whether a token w in x is a clue phrase or not (*auxiliary task*). Unlike obtaining annotations for counterfactual statements in the target language, it is relatively easier to obtain a list of counterfactual clue phrases for the target language. More importantly, as we show later in § 3.1, it is possible to automatically extract an accurate set of target language clue phrases, $\mathcal{V}_t$, using $\mathcal{D}_s$, $\mathcal{D}_{mt}$ and a set of clue phrases for the source language, $\mathcal{V}_s$.

The auxiliary task is motivated by prior work in semi-supervised learning (Ando and Zhang, 2005) and masked language modelling (Devlin et al., 2019), where it has been shown that by predicting tokens that are highly related (i.e. clue phrases) to the downstream task (i.e. sentence-level counterfactual detection) we can learn task-specific correlations between tokens. In contrast to the main task, which is modelled as a sentence-level binary classification task, we model the auxiliary task as a token-level sequence labelling task. However, unlike for the main task, where we have at least counterfactual labelled sentences from the source language (i.e. $\mathcal{D}_s$), we do not have any manually annotated training data neither for the source nor for the target languages for the auxiliary task. For this reason, we automatically label training data for the auxiliary task as follows. For the source language, we assign a binary-valued token label $y(w_j)$ for each token w_j in each sentence x_i in $\mathcal{D}_s$, where $y(w_j) = 1$, if $w_j \in \mathcal{V}_s$ and $y(w_j) = 0$ otherwise. For example, given a sentence "*The bottom fits fine, but I wished there was more ruching like in the photo.*" we label "*wished*", corresponding to a clue phrase in English as 1 and other tokens as 0. To generate training data for the target language we can use either sentences in $\mathcal{D}_t$ or $\mathcal{D}_{mt}$. We empirically compare the different combinations of training data later in §5.1.

Next, we describe the training objectives associated with the main and auxiliary tasks. Let us consider a multilingual masked language model (MLM), h, with pretrained parameters $\boldsymbol{\theta}$ that assigns a vector $h(w, x; \theta)$ to a word w in a sentence x. We train a feed forward neural network f with parameters $\boldsymbol{\phi}$ and a sigmoid output unit such that given the embedding $\boldsymbol{x}$ of a sentence x it predicts whether x is counterfactual (i.e. $f(\boldsymbol{x}; \boldsymbol{\phi}) = 1$) or otherwise (i.e. $f(\boldsymbol{x}; \boldsymbol{\phi}) = 0$). Different methods can be used to create sentence embeddings from MLMs such as mean or max pooling, attention-based weighting or by simply considering the embedding for the classification (i.e. [CLS]) token (Devlin et al., 2019). In our preliminary investigations we found that considering the [CLS] token embedding as a sentence representation to produce the best cross-lingual CFD performance despite its simplicity. However, we note that our proposed method is independent of the choice of the sentence encoder and can be combined with more complex sentence encoder architectures. In the subsequent discussion we denote $\boldsymbol{x} = h([\text{CLS}], x; \boldsymbol{\theta})$. Given that our main task of CFD is modelled as a binary classification task, the negative log-likelihood (NLL) loss for this prediction task can be written as in (1).

$$L_{\text{cfd}}(\mathcal{D}) = -\sum_{x \in \mathcal{D}} \big[(1 - y(x))(\log(f(\boldsymbol{x})) - 1) + y(x)\log(f(\boldsymbol{x}))\big] \tag{1}$$

For the auxiliary task, we train a feed forward neural network, $g(h(w, x); \boldsymbol{\psi})$, that takes in the contextualised embedding $h(w, x)$ of token w in sentence x and returns 1, if w is a clue phrase or 0 otherwise. We compute the NLL loss for the clue phrase prediction task as in (2).

$$L_{\text{cp}}(\mathcal{D}) = -\sum_{x \in \mathcal{D}} \sum_{j=1}^{|x|} \big[(1 - y(w_j))(1 - \log(z)) + y(w_j)\log(z)\big] \tag{2}$$

$$z = g(h(w_j, x); \boldsymbol{\psi})$$

Finally we add the losses for the main and auxiliary tasks to compute the total loss. Our zero-shot transfer model uses $\mathcal{D}_{mt}$ on the main task (1) and either of $\mathcal{D}_{mt}$ or $\mathcal{D}_t$ on the auxiliary task (2), i.e. $L_{\text{cfd}}(D_{mt}) + L_{\text{cp}}(D_t)$ or $L_{\text{cfd}}(D_{mt}) + L_{\text{cp}}(D_{mt})$ where we have dropped the model parameters for notational convenience. Further details on model training are provided in §4.

3.1 Automatic Clue Phrase Extraction

In some target languages such as low-resource languages, it might be even challenging to obtain a sufficiently large list of clue phrases covering various constructions used to express counterfactuality because of the difficulties in recruiting annotators. Moreover, in a true zero-shot spirit it is desirable not to assume any human supervision for the target language – neither for the main nor auxiliary tasks. Therefore, in this section, we propose a method to automatically extract clue phrases for the target language using the list of clue phrases for the source language, $\mathcal{V}_s$, counterfactual labelled dataset for the source language, $\mathcal{D}_s$, and its machine translated version, $\mathcal{D}_{mt}$. First, we use Awesome Aligner (Dou and Neubig, 2021), an off-the-shelf neural alignment model, and compute the alignment between each sentence $x_{s,i}$ in $\mathcal{D}_s$ and its translation $x_{mt,i}$. Next, for a token w_s in $x_{s,i}$, which is a clue phrase in the source language (i.e. $w_s \in \mathcal{V}_s$), we find the list of target language tokens, $\mathcal{A}_t(w_s)$. aligned with w_s in all sentence pairs, $\forall_{i=1}^{|\mathcal{D}_s|}(x_{s,i}, x_{mt,i})$. The candidate clue phrases in $\mathcal{A}_t(w_s)$ are further filtered following three criteria as described below. The end-to-end pipeline for target language clue phrase extraction is illustrated in Figure 1b between English (source) and Japanese (target) languages. To differentiate from the human annotated clue phrases (referred to as *gold* clue phrases here onwards), we call the clue phrases extracted via this alignment process as *auto-generated* clue phrases.

Criterion 1: Non-counterfactual Sentence Exclusion: Note that clue phrases can be ambiguous with regard to whether they express counterfactuality or not. For example, the clue phrase *wish* indicates a counterfactual statement in the sentence ***I wish** this shirt was available in red*, whereas it does not in *My **wish** came true*. Such ambiguous occurrences of counterfactual clues are likely to be aligned with non-counterfactual expressions in the target language. To reduce the noise due to this ambiguity in the unsupervised alignment process, we exclude non-counterfactual sentences from $\mathcal{D}_s$ and $\mathcal{D}_{mt}$ during the alignment process. In other words, we consider alignment between only sentence pairs $(x_{s,i}, x_{mt,i})$ such that $y(x_{s,i}) = 1$.

Criterion 2: Shared Term Exclusion: If a particular term w_t appears in candidate sets $\mathcal{A}_t(w_s)$ extracted for many distinct source language clue phrases w_s, it is likely that w_t is not a clue phrase but a high frequent functional word or a stop word. Therefore, we remove candidates appearing in more than one candidate set $\mathcal{A}_t(w_s)$ from target language clue phrase set.

Criterion 3: Majority Filtering: If a target language token w_t is aligned with the same source language clue phrase w_s in multiple sentence pairs, $(x_{s,i}, x_{mt,i})$, it increases the reliability of w_t as a clue phrase in the target language. We use this intuition to filter candidates, where for each source language clue phrase we select only the most frequently aligned target language token as a clue phrase. We refer to this filtering criterion as the *majority filtering*. In cases where there are multiple target language tokens with the same highest frequency of alignment with a specific source language clue phrase, we select all such tokens as target language clue phrases according to the majority filtering criterion.

	EN	DE	JA
Train	807 / 7,193	3865 / 1735	525 / 5,075
Dev	73 / 593	325 / 141	46 / 420
Test	150 / 1,184	650 / 284	96 / 838

Table 1: The number of sentences in AMCD with positive/negative label are shown respectively.

4 Experimental Settings

Dataset: We use the AMCD dataset, which contains counterfactual statements annotated from Amazon product reviews for three languages: English (EN), German (DE), and Japanese (JA). We use the original published training/development/test splits[2] in our experiments, for which the number of sentences are shown in Table 1. Throughout the experiments, we regard EN as the source language and DE and JA as the target languages. To create machine translated versions (i.e. $\mathcal{D}_{mt}$) of the EN dataset into the target languages, we use Amazon MT.[3]

Clue Phrase: The human annotated clue phrases provided by AMCD are considered as the gold clue phrases for each language. For the automatic clue phrase extraction described in §3.1, we use Awesome Aligner (Dou and Neubig, 2021) as the neural alignment model.

To evaluate the level of cross-lingual counterfactual detection (XCFD) performance that can be obtained by directly translating the source language clue phrases to the target language, we create a **Clue Phrase Translation** (CP Translation) baseline. This baseline uses Google Translate [4] to translate individual clue phrases in the source language to the target language without using any contexts for those clue phrases.

Models and Hyperparameters: To obtain token embeddings, we use two multilingual language models in our experiments:

[2] https://github.com/amazon-research/amazon-multilingual-counterfactual-dataset

[3] https://aws.amazon.com/translate/

[4] https://translate.google.com

Model	L_{cfd}	L_{cp}	DE		JA	
mBERT	$\mathcal{D}_t$		*90.3*	[88.1, 92.2]	*83.7*	[79.7, 87.3]
	$\mathcal{D}_s$		28.4	[26.0, 30.9]	47.3	[46.7, 47.8]
	$\mathcal{D}_{mt}$		70.9	[67.9, 73.8]	67.3	[62.7, 71.7]
	$\mathcal{D}_{mt}$	$\mathcal{D}_t$	**65.7**	[62.6, 68.7]	**68.6**	[64.6, 72.4]
	$\mathcal{D}_{mt}$	$\mathcal{D}_{mt}$	**73.0**	[70.1, 75.9]	**68.3**	[64.0, 72.4]
XLM-R	$\mathcal{D}_t$		*89.3*	[87.1, 91.4]	*86.2*	[82.4, 89.8]
	$\mathcal{D}_s$		45.1	[41.8, 48.3]	59.2	[53.8, 64.6]
	$\mathcal{D}_{mt}$		64.7	[61.7, 67.7]	81.1	[76.8, 84.9]
	$\mathcal{D}_{mt}$	$\mathcal{D}_t$	**68.0**	[65.1, 71.0]	**82.9**	[79.0, 86.6]
	$\mathcal{D}_{mt}$	$\mathcal{D}_{mt}$	**70.3**	[67.4, 73.3]	**81.9**	[77.6, 85.8]

Table 2: F1 scores on the test set of each target language with 95% confidence intervals in the brackets. The columns L_{cfd} and L_{cp} represent the dataset used respectively for the main (1) and auxiliary tasks (2). Models with blank L_{cp} are trained without the auxiliary task. The results of in-domain performance where labelled data from the target language is used to train a CDF model are shown in italics, the results with the auxiliary task are in shown bold face, and the best zero-shot result in each language is underlined.

mBERT (Devlin et al., 2019) and XLM-R (large model) (Conneau et al., 2019). Both of those models are transformer-based (Vaswani et al., 2017), but mBERT has been pretrained Wikipedia articles covering the 104 languages with the largest Wikipedias. On the other hand, XLM-R has been trained on 2.5TB of filtered CommonCrawl data containing 100 languages. The initial weights are taken from the `bert-base-multilingual-cased` and `xlm-roberta-large` model checkpoints, made available at the Huggingface transformers model hub (Wolf et al., 2020). We use the Adam optimizer (Kingma and Ba, 2014) with a batch size of 128, an initial learning rate of 0.00001 and train our CFD models for 5 epochs. As the evaluation metric, we report the macro-averaged F1 scores with 95% bootstrap estimated confidence intervals (Efron and Tibshirani, 1994).[5]

5 Results

5.1 Zero-shot Transfer with Auxiliary Task

Table 2 shows our main results of zero-shot cross-lingual transfer with the auxiliary task L_{cp} (2) together with the main task L_{cfd} (1). As an upper bound on performance, we train a CFD model using labelled data for the target language $L_{\text{cfd}}(\mathcal{D}_t)$ with mBERT and XLM-R separately. Recall that in the zero-shot setting we consider in this paper, we will not have access to such counterfactual labelled sentences for the target language. As a comparison, we report baselines, which are models trained on the main task with the source $L_{\text{cfd}}(\mathcal{D}_s)$ or the translation $L_{\text{cfd}}(\mathcal{D}_{mt})$ without the auxiliary task. We see that the best zero-shot cross-lingual transfer results are obtained using our proposed method for mBERT as well as XLM-R for both DE and JA. Specifically, F-score for DE improves from 70.9 to 73.0 in mBERT and for JA it improves from 81.1 to 82.9 in XLM-R by adding the auxiliary task to the main task on $\mathcal{D}_{mt}$. This supports our proposal to use clue phrase prediction in the target language as an auxiliary task for cross-lingual CFD.

L_{cp}	Clue Phrase Type	DE		JA	
$\mathcal{D}_t$	Human	66.9	[64.0, 69.9]	79.0	[76.0, 83.7]
	CP Translation	67.4	[64.4, 70.3]	79.0	[74.9, 82.7]
	Alignment	64.9	[61.8, 68.0]	80.1	[75.9, 83.9]
	Alignment^{1}	66.0	[63.0, 69.0]	**82.9**	[79.0, 86.6]
	Alignment^{2}	62.9	[59.9, 66.0]	77.5	[73.4, 81.4]
	Alignment^{3}	**68.0**	[65.1, 71.0]	79.5	[75.4, 83.3]
	Alignment^{1,2}	58.5	[55.4, 61.7]	79.1	[74.6, 83.2]
	Alignment^{2,3}	65.5	[62.5, 68.5]	80.0	[75.7, 83.9]
	Alignment^{1,2,3}	63.4	[60.3, 66.5]	78.9	[74.7, 82.7]
	Alignment^{1,3}	65.8	[62.7, 68.9]	80.4	[76.3, 84.1]
$\mathcal{D}_{mt}$	Human	**70.3**	[67.4, 73.3]	81.6	[77.4, 85.6]
	CP Translation	64.2	[61.1, 67.2]	81.3	[77.1, 85.3]
	Alignment	65.8	[62.9, 68.8]	81.6	[77.4, 85.5]
	Alignment^{1}	68.0	[65.0, 70.9]	79.5	[75.5, 83.4]
	Alignment^{2}	67.4	[64.4, 70.3]	81.7	[77.4, 85.6]
	Alignment^{3}	66.8	[63.7, 69.7]	**81.9**	[77.6, 85.8]
	Alignment^{1,2}	64.2	[61.1, 67.2]	78.2	[73.8, 82.2]
	Alignment^{2,3}	65.3	[62.3, 68.3]	75.7	[70.5, 80.5]
	Alignment^{1,2,3}	63.2	[60.1, 66.3]	78.4	[73.8, 82.4]
	Alignment^{1,3}	65.3	[62.3, 68.3]	79.6	[75.6, 83.3]

Table 3: F1 scores of XLM-R trained along different clue phrase types. All the scores are evaluated on the test set of each target language with 95% confidence intervals shown in the brackets. The filtering criteria used in each alignment approach is noted in its superscript. The best results in each language and L_{cp} are in bold face.

From Table 2 we see that among the models trained with the auxiliary tasks, XLM-R-based models ($L_{cp} \in \{\mathcal{D}_{mt}, \mathcal{D}_t\}$) perform better for JA than those obtained with mBERT, while the best model for DE ($L_{cp} = \mathcal{D}_{mt}$) is obtained using mBERT. In particular the best performance for JA is obtained with XLM-R (82.9), which is significantly better than the best performance for

[5] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.bootstrap.html`

L_{cp}	Clue Phrase Type	DE		JA	
$\mathcal{D}_t$	Human	63.7	[60.6, 66.8]	65.3	[61.6, 69.2]
	CP Translation	61.9	[58.7, 65.0]	66.4	[62.4, 70.2]
	Alignment	63.6	[60.5, 66.7]	64.7	[61.0, 68.6]
	Alignment^{1}	63.7	[60.5, 66.7]	61.9	[58.3, 65.5]
	Alignment^{2}	63.7	[60.6, 66.7]	**68.6**	[64.6, 72.4]
	Alignment^{3}	64.4	[61.3, 67.4]	66.4	[62.4, 70.3]
	Alignment^{1,2}	62.6	[59.5, 65.7]	**68.6**	[64.6, 72.4]
	Alignment^{2,3}	64.3	[61.2, 67.3]	66.1	[62.2, 69.9]
	Alignment^{1,2,3}	65.2	[62.1, 68.3]	65.4	[61.6, 69.3]
	Alignment^{1,3}	**65.7**	[62.6, 68.7]	67.6	[63.5, 71.5]
$\mathcal{D}_{mt}$	Human	70.7	[67.8, 73.7]	**68.3**	[64.0, 72.4]
	CP Translation	71.6	[68.7, 74.5]	**68.3**	[63.9, 72.4]
	Alignment	70.8	[67.9, 73.8]	65.8	[61.1, 70.2]
	Alignment^{1}	70.8	[67.8, 73.7]	67.0	[62.3, 71.5]
	Alignment^{2}	70.0	[67.0, 73.0]	64.9	[60.1, 69.5]
	Alignment^{3}	72.7	[69.8, 75.6]	66.4	[61.9, 70.7]
	Alignment^{1,2}	71.7	[68.7, 74.6]	66.2	[61.6, 70.4]
	Alignment^{2,3}	72.5	[69.6, 75.4]	66.7	[62.1, 70.9]
	Alignment^{1,2,3}	**73.0**	[70.1, 75.9]	65.1	[60.4, 69.5]
	Alignment^{1,3}	72.4	[69.5, 75.3]	64.5	[59.8, 69.0]

Table 4: F1 scores of mBERT trained along different clue phrase types. All the scores are evaluated on the test set of each target language with 95% confidence intervals shown in brackets. The filtering criteria used in each alignment approach is noted in its superscript. The best results in each language and L_{cp} are in bold face.

JA obtained with mBERT (68.6). Although the best performance for DE obtained with mBERT (73.0) is better than that with XLM-R (70.3), the performance difference between these two results are *not* statistically significant as evident from the overlapping confidence intervals. Note that compared to mBERT, which is trained on Wikipedias for different languages, XLM-R is trained on a much larger CommonCrawl corpus. Moreover, JA Wikipedia (530M tokens) is significantly smaller than that of DE (10297M tokens). Because mBERT tokenises CJK languages into individual characters and uses a 110K shared WordPiece vocabulary, the coverage of Japanese (which has lower overlap of subtokens with other languages) is less in mBERT. Therefore, XLM-R is capable of learning better representations for Japanese than mBERT, leading to better XCFD performance for JA.

In terms of the datasets used for the auxiliary task, the best model in DE uses the translation $\mathcal{D}_{mt}$, while that in JA uses the target corpus $\mathcal{D}_t$ for both mBERT and XLM-R. In general, the translation from EN to JA is harder than that from EN to DE as reported in Aiken (2019). Therefore, it is better to use $\mathcal{D}_t$ for the auxiliary task instead of $\mathcal{D}_{mt}$ when the translation quality for the target language is low such as from English to Japanese. Considering that $\mathcal{D}_{mt}$ is already used for the main task, by using $\mathcal{D}_t$ for the auxiliary task, which provides additional information not available by simply machine translating the sentences from the source language, we can provide extra supervision to the model.

Model	L_{cfd}	L_{cp}	DE		JA	
mBERT	$\mathcal{D}_{mt}$	$\mathcal{D}_t$	63.9	[60.7, 66.9]	68.1	[64.1, 72.0]
		$\mathcal{D}_{mt}$	73.1	[70.2, 76.0]	67.3	[63.0, 71.4]
XLM-R	$\mathcal{D}_{mt}$	$\mathcal{D}_t$	63.5	[60.5, 66.6]	79.7	[75.5, 83.4]
		$\mathcal{D}_{mt}$	68.8	[65.8, 71.6]	77.5	[73.5, 81.2]

Table 5: F1 scores of models trained on both of the human annotated and the automatically extracted clue phrase (the best clue phrase type shown in Table 2 is used). All the scores are evaluated on the test set of each target language with 95% confidence intervals shown in brackets.

5.2 Effect of Clue Phrase Choices

Table 3 and Table 4 show the results when using respectively XLM-R and mBERT as the text encoders with different clue phrase types including the human annotation, clue phrase translation, and our proposed alignment-based method (see §4 for detailed setting). The alignment-based method optionally has the three criteria described in §3.1 for filtering clue phrase candidates in the target language: **1** (*non-counterfactual sentence exclusion*), **2** (*shared term exclusion*), and **3** (*majority filtering*). We evaluate all possible combinations of filtering methods with the **Alignment** method, indicated by superscripts in Table 3. Alignment without any superscripts correspond to applying none of the candidate filtering criteria. Note that the results of XLM-R with the auxiliary task in Table 2 are the best results within each target language in Table 3.

From Table 3, we see that our alignment-based clue phrases can outperform manual clue phrases in both of $L_{\text{cp}}(\mathcal{D}_t)$ and $L_{\text{cp}}(\mathcal{D}_{mt})$ in JA, and $L_{\text{cp}}(\mathcal{D}_t)$ in DE with the best configuration. Furthermore, the best alignment-based clue phrases are better than clue phrase translation, which is still competitive compared to the human annotated clue phrases. This shows that high quality clue phrases can be automatically extracted using the method described in § 3.1. We reemphasize that

clue	context
wäre (would be)	dieses Produkt wäre toll, wenn... (This product would be great if ...)
wünschte (wished)	ich wünschte dieses Produkt wäre ...(I wish this product was ...)
hätte (would have)	hätte dieses Produkt ... (would this product ...)
könnte (could)	dieses Produkt könnte besser sein, wenn es ... (this product could be better if it ...)
と思っていた (thought it was)	Mサイズだと思っていた (thought it was M-size)
希望 (hope)	プラスチック製の物を希望していた (hope it was made from plastic)
かもしれない(could)	もう少し小さければ良かったかもしれない (could be better if it was smaller)
があれば(if it had)	蓋があれば良かった (if it had a lid)

Table 6: Automatically extracted clue phrases and their contexts for German (top) and Japanese (bottom) target languages. English translations are shown in brackets.

it is beneficial to be able to automatically extract clue phrases in zero-shot adaptation, because we might not always be able to recruit human annotators to manually compile clue phrase lists for all the target languages we would like to adapt to.

Table 4 shows the level of performance the proposed method would obtain if mBERT was used as the text encoding model. We see that the best performance for DE (73.0) is obtained by applying all filtering criteria, whereas with XLM-R the best performance for DE (70.3) was obtained with human-written clue phrases. However, as explained previously in § 5.1, the differences between these two results are not statistically significant. On the other hand, for JA we see that mBERT results are consistently lower than the corresponding XLM-R results across all filtering settings considered in Table 4 and Table 3. This comparison shows that the multilingual MLM used to encode text is an important choice for the performance of XCFD. However, this choice has been largely overlooked in prior work. For example, O'Neill et al. (2021) used only a single multilingual MLM (i.e mBERT only) in their cross-lingual evaluations. Although their reported best XCFD results with mBERT for JA and DE is better than those with our mBERT results, these results cannot be directly compared because unlike our zero-shot approach that does *not* use any labelled data for the target language, O'Neill et al. (2021) proposed a fully-supervised method where they use *all* of the available labelled data for the target language.

5.3 Combining Automatic Clue Phrase with Human Annotated Clue Phrase

We study the effect of incorporating both types of clue phrases (human annotated and automatically extracted) in the training process for the auxiliary task in Table 5. Compared to the best performances reported in Table 2 and Table 3 using the automatically extracted clue phrases, we see no further gains (in some cases even a drop) in performances for the target languages when using human annotated clue phrases in addition to the automatically extracted clue phrases in the auxiliary task. This shows that automatically extracted clue phrases are of a higher quality than the human-written clue phrases, and already capture the counterfactual clues contained in the human-written gold clue phrases. Some example clue phrases automatically extracted by the method described in § 3.1 are shown in Table 6 for German and Japanese target languages. We see that informative clue phrases are extracted by the proposed method for both of those target languages.

6 Conclusion

We studied zero-shot cross-lingual transfer learning for CFD and proposed a novel training objective that combines (a) token-level clue phrase prediction in target language sentences and (b) sentence-level counterfactuality prediction for source language (and translated to target language) sentences. Moreover, we proposed a method to automatically extract clue phrase for a given target language, which obviates the need for manually compiled clue phrases. Predicting clue phrases as an auxiliary task improves cross-lingual transfer from English source to German and Japanese target languages, obtaining state-of-the-art performances on AMCD.

7 Ethical Considerations

In this section, we discuss the ethical considerations related to these contributions. With regard to the dataset, we use the AMCD where the sentences were selected from a publicly available Amazon product review dataset. We do not collect or release any additional product reviews not included in the original AMCD as part of this paper. Although the dataset is manually verified that the sentences in the dataset do not contain any customer sensitive information, product reviews can contain socially biased opinions. However, we do not apply any bias mitigation methods in this paper, thus it is possible that the dataset biases present (if any) in AMCD are also encoded in the models we train in this paper. We use two pretrained multilingual language models, mBERT and XLM-RoBERTa, to obtain cross-lingual zero-shot CFD models. Those pretrained language models are known to be biased due to the curated pretraining corpus from web (Bommasani et al., 2020). Likewise for the dataset, we do not filter such social biases in the the language models. Therefore, we recommend that further evaluations to be performed before deploying the CFD models we train in this paper in real-world NLP systems.

References

Milam Aiken. 2019. An updated evaluation of google translate accuracy. *Studies in linguistics and literature*, 3(3):253–260.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Yang Bai and Xiaobing Zhou. 2020. Byteam at semeval-2020 task 5: Detecting counterfactual statements with bert and ensembles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 640–644.

Rishi Bommasani, Kelly Davis, and Claire Cardie. 2020. Interpreting Pretrained Contextualized Representations via Reductions to Static Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4758–4781, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised Cross-lingual Representation Learning at Scale. *arXiv preprint arXiv:1911.02116*.

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

Xiao Ding, Dingkui Hao, Yuewei Zhang, Kuo Liao, Zhongyang Li, Bing Qin, and Ting Liu. 2020. Hitscir at semeval-2020 task 5: Training pre-trained language model with pseudo-labeling data for counterfactuals detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 354–360.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics.

Bradley Efron and Robert J Tibshirani. 1994. *An introduction to the bootstrap*. CRC press.

Martin Fajcik, Josef Jon, Martin Docekal, and Pavel Smrz. 2020. BUT-FIT at SemEval-2020 task 5: Automatic detection of counterfactual statements with deep pre-trained language representation models. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 437–444, Barcelona (online). International Committee for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Anthony Janocko, Allegra Larche, Joseph Raso, and Kevin Zembroski. 2016. Counterfactuals in the language of social media: A natural language processing project in conjunction with the world well being project. Technical report, University of Pennsylvania.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yaojie Lu, Annan Li, Hongyu Lin, Xianpei Han, and Le Sun. 2020. Iscas at semeval-2020 task 5: Pre-trained transformers for counterfactual statement modeling. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 658–663.

Bella K Milmed. 1957. Counterfactual statements and logical modality. *Mind*, 66(264):453–470.

Anirudh Anil Ojha, Rohin Garg, Shashank Gupta, and Ashutosh Modi. 2020. Iitk-rsa at semeval-2020 task 5: Detecting counterfactuals. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 458–467.

James O'Neill, Polina Rozenshtein, Ryuichi Kiryo, Motoko Kubota, and Danushka Bollegala. 2021. I wish i would have loved this one, but i didn't–a multilingual dataset for counterfactual detection in product reviews. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 7654–7673, Online. Association for Computational Linguistics.

Youngseo Son, Anneke Buffone, Joe Raso, Allegra Larche, Anthony Janocko, Kevin Zembroski, H Andrew Schwartz, and Lyle Ungar. 2017. Recognizing counterfactual thinking in social media texts. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 654–658, Vancouver, Canada. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Len Yabloko. 2020. Ethan at semeval-2020 task 5: Modelling causal reasoning in language using neuro-symbolic cloud computing. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 645–652.

Xiaoyu Yang, Stephen Obadinma, Huasha Zhao, Qiong Zhang, Stan Matwin, and Xiaodan Zhu. 2020. SemEval-2020 task 5: Counterfactual recognition. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 322–335, Barcelona (online). International Committee for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.