

RECOGNITION OF COUNTERFACTUAL STATEMENTS IN TURKISH

**A Thesis Submitted to
the Graduate School of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of**

MASTER OF SCIENCE

in Computer Engineering

**by
Ali ACAR**

**July 2023
İZMİR**

We approve the thesis of **Ali ACAR**

Examining Committee Members:

Asst. Prof. Dr. Emrah İNAN

Department of Computer Engineering, Izmir Institute of Technology

Assoc. Prof. Dr. Kaya OĞUZ

Department of Computer Engineering, İzmir Economy University

Assoc. Prof. Dr. Selma TEKİR

Department of Computer Engineering, Izmir Institute of Technology

18 July 2023

Assoc. Prof. Dr. Selma TEKİR

Supervisor, Department of Computer
Engineering
Izmir Institute of Technology

**Prof. Dr. Cüneyt F. BAZLA-
MAÇCI**

Head of the Department of
Computer Engineering

Prof. Dr. Mehtap EANES

Dean of the Graduate School of
Engineering and Sciences

ACKNOWLEDGMENTS

Firstly, I wish to express my sincere gratitude to my advisor, Selma Tekir, for her outstanding supervision. If she were not patient, encouraging and helpful that much, this thesis would not be possible. In addition, I extend my gratitude to Emrah İnan and Kaya Oğuz for kindly consenting to be part of my thesis defence jury.

I want to thank my family for their endless support and trust in me. I also appreciate my friends who motivated and supported me throughout this project, with a special mention to my dear friend, Ali Sidar Yılmaz.

I am grateful to everyone who assisted in labelling the dataset. Additionally, I wish to thank Melike Üzümlü and Faik Utkan Denizer for their invaluable insights on counterfactuals in the Turkish language, and Umut Demirhan for his priceless help in the data collection process.

"Last but not least, I want to thank me. I want to thank me for believing in me. I want to thank me for doing all this hard work. I want to thank me for having no days off. I want to thank me for never quitting. I want to thank me for always being a giver and trying to give more than I receive. I want to thank me for trying to do more right than wrong. I wanna thank me for just being me at all times."

ABSTRACT

RECOGNITION OF COUNTERFACTUAL STATEMENTS IN TURKISH

Counterfactual statements describe an event that did not happen or cannot happen, and optionally the consequence of this event if it would happen. Counterfactual statements are the building blocks of human thought processes as people constantly reflect upon past happenings and consider their future implications. Counterfactual reasoning is essential for machine intelligence and explainable artificial intelligence studies. Detecting counterfactuals automatically with machine learning algorithms is very crucial for these areas.

This thesis presents the development of the first-ever Turkish counterfactual detection dataset. It presents a comprehensive classification baseline and expands the scope of counterfactual detection to include the Turkish language.

ÖZET

TÜRKÇE'DE KARŞIOLGUSAL İFADELERİN TANINMASI

Karşılgusal ifadeler, gerçekleşmemiş veya gerçekleşemeyecek bir olayı ve isteğe bağlı olarak bu olayın gerçekleşmesi durumundaki sonucu bildirir. Karşılgusal ifadeler, insan düşünce sürecinin yapı taşlarıdır. İnsanlar sürekli olarak geçmiş olayları ve bunların gelecekteki sonuçlarını düşünürler. Karşılgusal akıl yürütme, yapay zeka ve açıklanabilir yapay zeka çalışmalarında sıkça kullanılır. Makine öğrenmesi algoritmaları ile karşılgusal ifadelerin otomatik olarak tespit edilmesi bu alanlar için çok önemlidir.

Bu tez, Türkçe'deki ilk karşılgusal ifade tanıma veri kümesinin oluşturulmasını anlatmaktadır. Ayrıca, kapsamlı bir sınıflandırma deneyleri sunar ve karşılgusal ifade tespiti alanının kapsamını Türkçeyi de içerecek şekilde genişletir.

TABLE OF CONTENTS

LIST OF FIGURES	viii
LIST OF TABLES	ix
CHAPTER 1. INTRODUCTION	1
CHAPTER 2. BACKGROUND	4
2.1. Text Representation Background	4
2.1.1. Language Modelling	4
2.1.2. Neural Networks	4
2.1.3. Masking Language Modeling (MLM)	5
2.1.4. BERT	5
2.1.5. XLM-RoBERTa	6
2.2. Classification Methods	6
2.2.1. Transfer Learning	6
2.2.2. Zero-shot Cross-lingual Classification	7
2.3. Counterfactual Statement in Turkish	7
CHAPTER 3. RELATED WORKS	10
CHAPTER 4. DATASET	12
4.1. Data Collection	12
4.2. Annotation	14
4.3. Dataset Statistics	15
CHAPTER 5. EXPERIMENTS	28
5.1. The Effect of Clue Phrases	29
5.2. Cross-Dataset Performance	31
5.3. Combined Datasets Performance	32
5.4. Zero-shot Classification Results	34
5.5. TRCD Detailed Classification Results	35
CHAPTER 6. CONCLUSION	39
6.1. Limitations	40
6.2. Future Work	40

REFERENCES	41
APPENDICES.....	44
APPENDIX A. DETAILED TRCD DATASET STATISTICS	44
APPENDIX B. QUALITATIVE ANALYSIS OF THE CLASSIFICATION OF THE TRCD DATASET.....	46

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
Figure 4.1.	Similarity query option in TNC	18
Figure 4.2.	Similarity query result for <i>geleydi</i>	19
Figure 4.3.	PoS Affix query option in TNC	20
Figure 4.4.	PoS Affix query result for VB + avsa+vi+past+pagr	21
Figure 4.5.	Standard Query option in TNC	22
Figure 4.6.	Standard Query result for word <i>özgür</i>	23
Figure 4.7.	The annotation process	24
Figure 5.1.	The classification model.....	29

LIST OF TABLES

<u>Table</u>	<u>Page</u>
Table 1.1. Example sentences.....	1
Table 2.1. Turkish CFD Clue Phrases. VB refers to the verb. Other tags explanations are shown in Table 2.2.....	8
Table 2.2. TNC tag explanations.....	9
Table 4.1. Clue phrase distribution in dataset versions	15
Table 4.2. The two-phased annotated sentences as they are, with more context and undecided percentages of the annotations. The sentences are given as bold in the context.....	16
Table 4.3. Turkish counterfactual dataset statistics	17
Table 4.4. CFD datasets clue phrase distribution comparison.....	25
Table 4.5. CFD datasets comparison	26
Table 4.6. Domain statistics of the Turkish CFD dataset	27
Table 5.1. MCC and F1 (macro) results for datasets with mask and no mask options.....	30
Table 5.2. Cross-dataset Performance of the models with no mask option	31
Table 5.3. Results of combining dataset experiment. We used abbreviations for the datasets: E for AMCD-EN, D for AMCD-DE, J for AMCD-JP, S for Semeval and T for TRCD.....	33
Table 5.4. Zero-shot classification macro-F1 scores on the 5 CFD datasets.....	34
Table 5.5. Confusion Matrix of TRCD fine-tuned BERTurk model with no mask (Predicted _{nm}) and mask options (Predicted _m).....	35
Table 5.6. Confusion Matrix of (E + D + J + S) fine-tuned XLM-R model with no mask (Predicted _{nm}) and mask options (Predicted _m).....	36
Table 5.7. Classification results for the BERTurk model and Zero-shot model for each clue phrase of the TRCD dataset. Count refers to the sentences with the clue phrase in the TRCD dataset. Pos. % refers to the positive percentage of the sentences with the clue phrase. In the scores, N and P refer to negative and positive class F1 scores, respectively.	37

Table 5.8.	Classification results for the BERTurk model and Zero-shot model on each domain of the TRCD dataset. Count refers to the sentences with the domain in the TRCD dataset. Pos. % refers to the positive percentage of the sentences with the domain. In the scores, N and P refer to negative and positive class F1 scores respectively.	38
Table A.1.	Clue phrases statistics of the Turkish CFD dataset	44
Table A.2.	Random words statistics of the Turkish CFD dataset	45
Table B.1.	Qualitative Analysis for misclassified Positive class instances. The model which miss-classified the sentence given in the Models column. T: BERTurk, Z: the zero-shot model which fine-tuned XLM-R with (D + E + J + S) dataset combination, X: XLM-R model which fine-tuned with TRCD. The subscripts indicate masking strategy, <i>nm</i> for no masking and <i>m</i> for masking the clue phrases.	46
Table B.2.	Qualitative Analysis for misclassified negative class instances. The model which miss-classified the sentence given in the Models column. T: BERTurk, Z: the zero-shot model which fine-tuned XLM-R with (D + E + J + S) dataset combination, X: XLM-R model which fine-tuned with TRCD. The subscripts indicate masking strategy, <i>nm</i> for no masking and <i>m</i> for masking the clue phrases.	47

CHAPTER 1

INTRODUCTION

Counterfactual statements describe an event that did not happen or cannot happen, and optionally the consequence of this event if it would happen (Milmed, 1957). For example, the sentence *if it were raining, I would need an umbrella* (*yağmur yağsaydı, şemsiyeye ihtiyacım olacaktı*) describes an event that did not happen in the first part, also called **antecedent** and gives its possible consequence in the second part, also called **consequent**.

To distinguish between counterfactual and non-counterfactual statements, we must know about the event in the interest sentence. This knowledge is generally given in the sentence with the context. However, sometimes common-sense knowledge is sufficient. In Table 1.1, the first and second sentence gives the knowledge of the event with the context. In the first sentence, we understand that she/he continued reading, so there is a counterfactual statement describing the opposite in the first part. In the second sentence, there is a guess that the mother is sleeping. We understand this from the first part of the sentence. However, we cannot distinguish whether the sentence has a counterfactual statement in the third sentence since we do not know whether she/he has told everything to Selim and asked for his help. As the example sentences state, detecting counterfactual statements is a challenging task.

Table 1.1. Example sentences

Sentence	Context Type	Counterfactuality
Giriş bölümünün ardından okumaktan vazgeçebilirsiniz ama siz devam etmeyi seçtiniz. (You could have stopped reading after the introduction, but you chose to continue.)	Given	Counterfactual
Bütün ışıklar sönmüştü, annem çoktan yatmış olmalıydı. (All the lights were out, my mother must have gone to bed already.)	Given	Not counterfactual
Her şeyi Selim'e anlatıp ondan yardım istemeliydi. (She/He should have told everything to Selim and asked for his help.)	Not given	Indistinguishable

Counterfactual statements are the building blocks of human thought processes as people constantly reflect upon past happenings and consider their future implications.

David Hume (Hume and Millican, 2007) defines this way of human thinking as causation by the sentence, "If the first object had not been, the second never had existed.". Judea Pearl (Pearl and Mackenzie, 2018) places counterfactuals at the top of the ladder of causation and puts causal reasoning at the core of human intelligence. Thus, understanding counterfactuals is also a crucial ability for machine intelligence.

Counterfactual detection (CFD) is an essential task for natural language processing (NLP), and it has been studied in psychological assessment using social media texts ((Janocko et al., 2016), (Son et al., 2017)) and customer analysis with product reviews (O'Neill et al., 2021). CFD is framed as a sentence-level binary classification task and has been studied on texts of varying lengths in different domains and languages. (Son et al., 2017) performs CFD on social media texts, and the SemEval-2020 task (Yang et al., 2020) on news reports of finance, politics, and healthcare. (Yang et al., 2020) also addresses the problem of detecting antecedent and consequent. While these two works target CFD in English, (O'Neill et al., 2021) work with English, German, and Japanese e-commerce customer reviews. All the works accompany a human-annotated CFD dataset, as well. Another study (Ushio and Bollegala, 2022) proposes a novel zero-shot cross-lingual transfer learning solution for creating a CFD dataset in a new language with less human effort.

(Son et al., 2017) reported that counterfactual statements rarely exist in natural language, 1-2% in social media texts. This handy cap might cause creating an imbalanced dataset. All prior works filtered their collected sentences using a clue phrase list (e.g. *if*, *wish* in English). The filtering aims to increase the percentage of sentences with counterfactual statements to create a more balanced dataset.

Developing a CFD dataset for a new language is complex and has two main challenges: developing a language-specific clue phrase list and manual annotation of counterfactuality (Ushio and Bollegala, 2022). (O'Neill et al., 2021) have tried automatically creating a CFD dataset for German and Japanese languages using machine translation (MT) of English sentences. However, they state that using MT is unsuitable for creating a new CFD dataset. Also, to automatically develop a new language's clue phrase list and CFD dataset, (Ushio and Bollegala, 2022) proposed a novel zero-shot cross-lingual transfer method.

The contributions of this thesis are as follows: (1) We studied CFD for Turkish and in fictional text domain first time in literature. We publish a 5k-sized sentence-level human-annotated Turkish CFD dataset¹. (2) We train classifier models with our annotated dataset using various encoding methods and classifier algorithms to give a comprehensive baseline.

The remainder of this thesis is organized as follows. In section 2, we give the background. Section 3 describes the related work, including research on counterfactual

¹The dataset can be found at <https://github.com/dopc/turkish-counterfactual-recognition>

statements in different domains, CFD dataset works on various domains of text and classification methods on CFD task. In section 4, we describe details of the developing process of our CFD dataset. In section 5, we provide the CFD classification baseline with comprehensive experiments. Finally, section 6 presents the conclusion.

CHAPTER 2

BACKGROUND

2.1. Text Representation Background

Text representation is essential in natural language processing (NLP) tasks like text classification. It converts text into a numerical format that classification algorithms can handle.

2.1.1. Language Modelling

Language modelling is a key concept in the field of NLP. The core idea behind language modelling is to capture a language's underlying syntactic, semantic, and contextual properties with observed patterns in large-scale datasets. These models can serve as the base for various downstream NLP tasks, such as speech recognition, machine translation, text summarization, question-answering systems, and text classification.

There are different approaches to language modelling, including traditional n-gram models, which rely on counting the occurrence of word sequences, and the more recent deep learning-based methods, which use neural networks to learn complex representations of words and their contexts. One of the major breakthroughs in language modelling has been the development of powerful pre-trained models like Word2Vec (Mikolov et al., 2013b) and BERT (Devlin et al., 2019). These pre-trained models can be fine-tuned for specific tasks, reducing the need for large amounts of labelled data and significantly improving results in many NLP applications.

2.1.2. Neural Networks

In the area of language modelling and text representation, neural network-based approaches are powerful techniques to capture the semantic and syntactic information in textual data. These models are often referred to as distributed representations or word embeddings. They work by using deep learning algorithms to map words or phrases onto

continuous vector spaces, where semantically similar words are situated closer to each other. In order to achieve this, neural networks utilize large amounts of text data to train their models and subsequently learn word representations based on the words' contextual relationships with surrounding words.

Early implementation of these techniques is Word2Vec, significantly advancing NLP tasks such as sentiment analysis, machine translation, and text classification.

2.1.3. Masking Language Modeling (MLM)

Masking Language Modeling (MLM) is a training objective of the text representation method, especially with neural networks (NN). In MLM, words or tokens in a given text are randomly masked or replaced with a special token, such as [MASK]. The objective is to train a neural network language model to predict the masked tokens based on the surrounding words.

MLM is generally implemented with neural networks using architectures like transformers (Vaswani et al., 2017). In training, the network learns to predict the masked token(s). This process trains the model to learn the contextual information and dependencies between words, resulting in a better text representation. The MLM-based model represents each token as its learned features and contextual information. The text representation is obtained from an intermediate or the model's output layer.

2.1.4. BERT

BERT (Bidirectional Encoder Representations from Transformers) is a transformer architecture language model. The key strength of BERT lies in its ability to generate rich, contextualized text representations. By considering each word's context within the sentence bidirectionally, BERT captures a deeper understanding of the text compared to traditional methods, like TF-IDF.

BERT revolutionized text representation by introducing a pre-training and fine-tuning approach. In the pre-training, BERT is trained with the MLM method on massive amounts of unlabeled text. Once pre-trained, BERT can be fine-tuned for various downstream tasks, including text classification. During fine-tuning, BERT is trained on task-specific labelled data.

In addition to BERT, there is also a variant called Multilingual BERT (mBERT) (Devlin et al., 2019). mBERT is pre-trained on a corpus of multilingual text. This model extends the capabilities of BERT for multiple languages.

2.1.5. XLM-RoBERTa

XLM-RoBERTa (XLM-R), which stands for Cross-Lingual Masked RoBERTa is built on the RoBERTa (Liu et al., 2019) architecture which is another MLM-based pre-trained language model. RoBERTa outperforms BERT with its larger pre-training data and more effective training procedure. XLM-R is designed to perform well in various languages since it can learn common structures from the text in 100 different languages.

To compare XLM-R with BERT and mBERT, several differences can be found. Firstly, BERT is primarily focused on English language tasks, while mBERT is its multilingual version. XLM-R builds on the RoBERTa architecture, which outperforms BERT with improved training strategies and larger pre-training data. This results in better performance and generalization capabilities.

XLM-R's most important advantage over mBERT lies in its pre-training mechanism and capacity to learn from a more diverse range of languages, causing better performance in cross-lingual tasks and a better ability to transfer information across languages.

2.2. Classification Methods

2.2.1. Transfer Learning

Transfer learning is using a pre-trained model like BERT leveraging the knowledge learned from its pre-training phase and applying it to a downstream task. This process is often referred to as fine-tuning.

A classification layer is added on top of the pre-trained model to utilise the model for a classification task. This additional layer enables the model to map the contextualized representations generated by the pre-trained model to the specific classes or categories of the classification task.

During fine-tuning, the parameters of the pre-trained model are usually updated with the parameters of the classification layer. This allows the model to adapt the pre-trained representations to the target task by fine-tuning the weights based on the task-specific data.

Fine-tuning minimizes the loss of the classification layer. The objective is to minimize the difference between the predicted class probabilities and the true labels.

Transfer learning with a pre-trained model also decreases computation and training time since the model is not trained from scratch. Instead, it starts with pre-trained

representations that have learned a huge amount of text data.

2.2.2. Zero-shot Cross-lingual Classification

Zero-shot cross-lingual text classification is an approach in NLP that allows text classification in different languages without requiring language-specific training data. This method is highly beneficial because it accelerates the development of multi-language NLP systems, which traditionally involve extensive data annotation and model training for each target language.

In zero-shot cross-lingual text classification, a model is trained only in one source language but can predict the class labels in a different target language. It does that by using the shared knowledge between languages. This method often relies on multilingual word embeddings or transformer-based architectures like BERT, which depends on capturing semantic similarity across languages. However, this approach still presents challenges, such as effectively transferring knowledge from high-resource to low-resource and dealing with typological language diversities.

2.3. Counterfactual Statement in Turkish

We utilized the counterfactual statement definition proposed by (Üzüm, 2020) for the Turkish language. This work defines counterfactual statements as descriptions of unreal situations that run parallel to a specific reality. (Üzüm, 2020) classifies counterfactual statements into three types based on usage and provides common linguistic structures for each type, as outlined below.

- **Counterfactual wishes** Wishing a present or past situation to be different from the current situation. They gave *-sA*, *-(s)AyDI*, *-(s)AymIş*, *keşke* linguistic structures as common indicators for this type of counterfactual statement. (Example: *Keşke kendine ait bir evin olsaydı* (Wish you had your own house))
- **Counterfactual conditions** Conditionally creating a situation different from the current situation in the present or in the past. They gave *-(s)AyDI*, *-ArdI*, *-mAzDI*, *eğer* linguistic structures as common indicators for this type of counterfactual statement. (Example: *Eğer seneler önce iflas etmeseydin, şu an kendine ait bir evin olurdu*. (If you hadn't gone bankrupt years ago, you would have a house of your own now.)
- **Other counterfactuals**

- To express that a situation different from what is planned or expected to happen has occurred. They gave **-AcAktI**, **-ArdI**, **yoksa** linguistic structures as common indicators for this type of counterfactual statement. (*Example: 2015'te kendi evin olacaktı, 2009'da iflas etmen buna engel oldu (You would have your own house in 2015, your bankruptcy prevented this in 2009)*)
- To describe a situation different from the current situation when declaring a belief or possibility. They gave **-mAllydI** linguistic structures as common indicators for this type of counterfactual statement. (*Example: 2009 krizini daha iyi yönetmeliydin (You should have handled the 2009 crisis better)*)

In previous CFD studies, linguistic structures like the above are defined as clue phrases in CFD task in various languages (e.g. *would* for English). In these studies, a clue phrase list was created to filter candidate sentences during dataset creation. To the best of our knowledge, there is no existing CFD clue phrase list for the Turkish language. Therefore, we developed one based on the linguistic structures presented in (Üzüm, 2020). Table 2.1 illustrates our clue phrase list with their TNC equivalents. **VB** refers verb and the other tags are depicted in Table 2.2 with their TNC explanations¹.

All existing CFD studies used the counterfactual definition provided by (Janocko et al., 2016), and most of them created their own clue phrase list. We preferred a Turkish-specific counterfactual study due to the unique morphological structure of Turkish. All of the clue phrases we used for the Turkish language were parts of the verb inflections. As (Denizer, 2023) indicated, the meanings of Turkish clue phrases highly rely on context. Therefore, specifying Turkish clue phrases is a linguistically challenging problem.

Table 2.1. Turkish CFD Clue Phrases. **VB** refers to the verb. Other tags explanations are shown in Table 2.2

Clue Phrase	TNC Equivalent	# in TNC
-sA	VB + avsa+pagr	4624
-ArdI	VB + aor+vi+past+pagr	2637
-AcAktI	VB + futr+vi+past+pagr	2169
-AyDI	VB + avsa+vi+past+pagr	2164
-AbilArdI	VB + abil+aor+vi+past+pagr	1678
-mAzdI	VB + neg+aor+vi+past+pagr	1676
-AmAzdI	VB + abil+neg+aor+vi+past+pagr	1370
-mAllydI	VB + necc+vi+past+pagr	1220
-sAlArDI	VB + avsa+pagr+vi+past	475
-Aymlş	VB + avsa+vi+perf+pagr	123

¹<https://v3.tnc.org.tr/help/tagset-for-part-of-speech-and-affixes>

Table 2.2. TNC tag explanations

Tag	Morpheme	Function	As in
abil	A, Abil	auxiliary verb	gelemez, gelebilir
aor	Ø, r, z	TAM_aorist	acımayız, uyursun, uyumaz
avsa	sA, A	adverbial	gitse, gideydi
futr	AcAk	TAM_future	gidecek, gideceklerden
necc	mAll	TAM_necessity	gitmeli
neg	mA	negative	gitmedik
pagr	Ø, m, (n)Iz	person agreement	gittim, gitti, gittiniz
past	DI	TAM_past / perfective	gitti
perf	mIş	TAM_referential/perfective	gitmiş
vi	i	verb	<i>gittiyse</i>

CHAPTER 3

RELATED WORKS

In one of the earliest CFD works (Son et al., 2017), they modelled the CFD task as a binary classification problem and collected social media texts from Twitter using a clue phrase list. Trained human annotators manually annotated tweets and created the first CFD dataset. In experiments, they used rule-based methods and statistical models for classification.

In SemEval2020, Task5 (Yang et al., 2020) was Counterfactual Recognition. This task consisted of two different subtasks. The first subtask was a binary classification problem, a CFD task. They created a dataset (we refer to as **Semeval**) using sentences from news reports on finance, politics, and healthcare. They used a clue phrase list and tag-based filtering while collecting the sentences. Then, human annotators manually labelled the sentences and developed the dataset.

Several participants of this subtask (Ding et al., 2020; Fajcik et al., 2020; Lu et al., 2020; Ojha et al., 2020; Yabloko, 2020) used various pre-trained transformer based neural models and got the top-ranked results. Some participants (Ojha et al., 2020) used classic machine learning methods like SVM and got lower accuracy.

The CFD studies mentioned above are only in the English language. The first multilingual work (O’Neill et al., 2021) studied the CFD task in three different languages: English, German and Japanese. The Amazon Multilingual Counterfactual Dataset (AMCD) is released. They used customer reviews from Amazon Customer Reviews Dataset¹. They created a clue phrase list for each language and used it for filtering the collected reviews. They also used random reviews which do not include clue phrases to avoid introducing a selection bias towards the clue phrases. They used various text representation methods and classification algorithms and shared their results as a baseline for all three languages.

They also experiment with cross-lingual classification using machine translation (MT). To do that, they trained classification models with the English dataset. Then, they translated German and Japanese sentences to English with Amazon MT². They classified these translated sentences with the model which is trained with the English dataset. The results were significantly lower than those obtained using models trained with their respective languages (e.g., classifying German sentences with a model trained on the German dataset). So, the results showed that MT translation is not a suitable solution for

¹<https://s3.amazonaws.com/amazon-reviews-pds/readme.html>

²<https://aws.amazon.com/translate/>

classifying sentences from different languages.

A recent work (Ushio and Bollegala, 2022) proposed a novel method for zero-shot cross-lingual classification for the CFD task. In this method, they used a CFD dataset of a language (they call it *source language*) to automatically build a clue phrase list and a CFD model for another language (they call it *target language*), which had not been studied before. The proposed method achieved a significant accuracy increase in cross-lingual transfer from English to German and English to Japanese on AMCD.

CHAPTER 4

DATASET

4.1. Data Collection

In this work, we followed (Üzümlü, 2020) and used Turkish National Corpus (TNC) (Aksan et al., 2012) as the data source. TNC is a publicly available Turkish corpus. It consists of fictional and informational texts collected from various sources (e.g. news, forums, literary) and different mediums (i.e. spoken and written). We have used all sources and written medium texts for our data collection process.

TNC has a query interface which takes the key phrase and returns the texts which include the queried key phrase. Since our clue phrases are not words, we used the **PoS Query (Affix)** option of TNC. First, we need to get the TNC equivalent of the clue phrase to do that. This can be done with an example word which contains the clue phrase. For example, for the clue phrase **-AyDI** we used *geleydi* with **Similarity Query** option of TNC in Figure 4.1. The result gives you the TNC equivalent of your query word, **VB + avsa+vi+past+pagr** as shown in Figure 4.2. After doing this for each clue phrase, we got the TNC equivalents of all clue phrases, as given in Table 2.1. We used the Turkish version of the TNC website, so the TNC equivalents in Table 2.1 differ from those in the figures. Finally, after getting all TNC equivalents of our clue phrases, we used **PoS Query (Affix)** query option of TNC, as shown in Figure 4.3 and Figure 4.4. Then, we used **Sentence View** and finally downloaded the sentences with **CSV** option.

In the data collection process, we mainly followed the data collection method of (O’Neill et al., 2021). Our data collection process consists of two main parts. In the first part, we used all ten clue phrases in Table 2.1 and queried them via the TNC query interface, as explained above, and downloaded the texts. We filtered out the texts which have more than one sentence using NLTK sentence tokenizer (Bird et al., 2009). We also filtered out some outlier sentences which is written in various Turkish dialects (e.g. *Tam ben diycektim. (I was just about to say.)*). We also filter out the texts with more than one clue phrase. To do that we used a morphological analyzer for Turkish TRmorph (Çöltekin, 2010). After that, we only kept the sentences with more than ten and less than 512 mBERT¹ and XLM-R² tokens. Too short and long sentences would be difficult for a

¹<https://huggingface.co/bert-base-multilingual-cased>

²<https://huggingface.co/xlm-roberta-base>

human to annotate since a too short sentence might not include enough information and a too long sentence might include unnecessary information besides CFs. As a result, we had 3500 sentences with exactly one clue phrase.

As stated in (O’Neill et al., 2021), using only sentences which include clue phrases may introduce a selection bias. They collected sentences without clue phrases to overcome this selection bias and used them in their dataset. We followed this method to overcome this bias and selected 30 Turkish words. We refer to them as random words to distinguish them better.

To make these 30 random Turkish words diverse, we used the formula from (Mikolov et al., 2013a) that is shown in 4.1. (Mikolov et al., 2013a) used this formula for the Negative Sampling method. In this formula, f is a function that returns the frequency of a word w_i . As (Mikolov et al., 2013a) stated, they experimented with various versions of this formula, and they got the best result with 3/4rd power of word frequency in the formula. In the denominator, the formula used the sum of the frequencies of all words. As a result, $P(w_i)$ gives a modified probability for the word w_i .

To use this formula in our problem, we got the Turkish words and their frequencies from a Python library wordfreq³ (Speer, 2022). In this library, the Turkish words are collected from Wikipedia, subtitles, the web (OSCAR⁴ (Ortiz Suárez et al., 2019)) and Twitter. Finally, we applied the formula to the word frequencies and selected 30 words with the highest probabilities. Then we filtered out the non-Turkish words and words which are shorter than three letters among these 30 words. In the end, 25 Turkish words survived, shown in Table A.2.

$$P(w_i) = \frac{f(w_i)^{3/4}}{\sum_{j=0}^n (f(w_j)^{3/4})} \quad (4.1)$$

We queried each of these 25 survived words on TNC with **Standard Query (Without PoS tag)** option, as shown in Figure 4.5 and Figure 4.6, and downloaded the texts. Again, we filtered out too long and too short texts again and texts with more than one sentence. Most importantly, we filtered out sentences that contain clue phrase(s). To do that we again used TRmorph.

Finally, we got a 3500 sentences sample from the second iteration to combine sentences from these two iterations. While doing that, we ensured that this sample has the same domain distribution as the sentences from the first iteration. This decision aims not to introduce a contextual bias because of the domain of the sentences, the domain distribution is shown in Table 4.6.

In the end, we got 3500 sentences with clue phrases and 3500 sentences without

³<https://github.com/rspeer/wordfreq>

⁴<https://oscar-project.org/>

clue phrases, a total of 7000 sentences. Finally, all sentences are manually annotated as described in Section 4.2..

4.2. Annotation

The annotation process is the step in which the dataset is labelled by human annotators with respect to the classification objective. We prepared guidelines⁵ to train our human annotators. The guidelines consist of definitions, extensive examples and counterexamples.

Annotators may still disagree on the same sentence even after being trained with the guidelines. In the annotation process, every research aims for a high annotation agreement between their annotators. Two main methods are applied in prior works. **(1)** Every text is annotated by more than one person and only highly agreed texts are accepted to exist in the labelled dataset. For example, (O’Neill et al., 2021) created their CFD dataset by ensuring that at least 90% of the sentences had agreement from 2 out of 2 annotators. A third annotator resolved any disagreements for the remaining 10% of sentences. **(2)** More than one person annotates a small percentage of the collected sentences. Then, Inter Annotator Agreement (IAA) score is calculated between these annotations. IAA score is expected to be high enough to say there is agreement among the annotators and that the annotation process is coherent.

This thesis followed the agreement method (2) to make our annotation process coherent. We formed groups of four annotators, with each group assigned to annotate 1000 sentences. In each group, 100 sentences were annotated by all four annotators. In the end, a total of 14% (1000/7000) of the sentences are annotated by four different annotators. For each group, we accept 3/4 and 4/4 agreed annotations. A fifth annotator resolved the disagreement of the 2/4 agreed annotations. The annotation process is illustrated in Figure 4.7.

After the annotation process, we discarded some sentences. Some of these sentences were not complete, and some were annotated as **undecided** by the annotators. We analyzed the undecided sentences and realized that most of these sentences do not have enough context to decide their classes. As (Denizer, 2023) pointed out, the meanings of Turkish clue phrases highly rely on context. Table 4.1 shows the distribution of the clue phrases in all annotated data which is 7k-sized, our 5k-sized final dataset and the filtered-out sentences since they are labelled as undecided. The results show that clue phrases **-mallydI** and **-AbilArdI** have significantly high undecided labelling percentages.

We conducted a qualitative analysis of the effect of the context in a sentence while

⁵The guidelines can be found at <https://github.com/dopc/turkish-counterfactual-recognition>

Table 4.1. Clue phrase distribution in dataset versions

Clue Phrase	% in 7k	% in 5k	% in Undec.
-mAlIydI	7.49	5.7	46.56
-AbilArdI	6.47	5.5	28.12
-ArdI	6.76	7.06	7.19
-AmAzdI	6.73	7.16	7.19
-mAzDI	5.56	5.92	5.94
-AydI	3.93	4.34	2.50
-sA	10.44	11.86	1.25
-sAlArDI	1.56	1.78	0.62
-AcAkDI	0.31	0.32	0.62
-AymIş	0.33	0.36	0.

deciding whether the sentence has a counterfactual statement. To do that, we collected ten sentences with some specific clue phrases (i.e. *-mAlIydI*, *-AbilArdI*, *-AcAkDI*) from TNC. We also got a longer version for each sentence with several sentences containing more context. All the annotators labelled these sentence pairs as they are and with more context. Table 4.2 shows the sentences, more context versions and undecided label percentages for both annotation phases. As shown in the table, undecided percentages decreased when the annotators saw more context. This result shows that one may need more context to distinguish the counterfactuality of a sentence in Turkish.

Finally, we got 5000 annotated sentences and measured 65.04 Fleiss’s kappa score (Fleiss, 1971) with the four times annotated sentences.

4.3. Dataset Statistics

In this thesis, we introduce the first-ever Turkish counterfactual dataset (TRCD). Basic statistics of the dataset can be found in Table 4.3. The TRCD comprises 5,000 Turkish sentences, of which 12.8% contain counterfactual statements. Half of the sentences include clue phrases, while the other half do not. We collected 2500 sentences with clue phrases in the first iteration of the dataset collection process. These sentences contain 24.6% positive examples. In the second iteration, we added 2500 sentences without clue phrases that contain 1% of counterfactual statements.

(O’Neill et al., 2021) is the only CFD work sharing its clue phrase lists and a dataset, AMCD. They curated clue phrase lists for three languages: English, German and Japanese. Table 4.4 shows distributions of clue phrases in positive and negative classes of AMCD and our dataset. German and English have a similar number of clue phrases, and Japanese has much more than them. We created the first-ever clue phrase list for the CFD problem for the Turkish language. We selected ten clue phrases among the linguistic

Table 4.2. The two-phased annotated sentences as they are, with more context and undecided percentages of the annotations. The sentences are given as bold in the context.

Sentence	Undecided Percentage	Sentence w/ more context	Undecided Percentage
İlk yudumdan sonra her şey başkalaşacaktı sanki.	15.79	Elindeki kadeh bir masal ikisiri duygusu veriyordu ona. İlk yudumdan sonra her şey başkalaşacaktı sanki. İyi ya da kötü, ama mutlaka başkalaşacaktı. Adam'ın yanında duyduğu kaynağı belirsiz o kör güvene rağmen, ilk yudumu çekinerek aldı ağzına Alice.	0.0
Şu ıslak çuvalların altında olmalıydı.	63.16	Ortalıkta balık görünmüyordu. Şu ıslak çuvalların altında olmalıydı. Çuvalları eliyle kaldırmak istedi. Esmer delikanlı yanına gelip engelledi: Balık malık yok kardeşim.	10.53
Mahmut Can yetiştirdiği tavukların yumurtalarını annemin hastanesine satabilirdi.	63.16	Aklıma çocuğa bahçede bir küçük tavuk çiftliği kurmak, onun hayvan sevgisine bir de ekonomi eğitimi katmak geldi. Mahmut Can yetiştirdiği tavukların yumurtalarını annemin hastanesine satabilirdi. Üsküdar Çarşısı'ndaki Tarım Kooperatifi'nden birkaç tane civciv aldık. İkisi yaşadı. Mahmut Can bunlardan birini çok sevdi odasına taşıdı, orada baktı, büyüttü.	0.0

structures and words which are given in (Üzüm, 2020) to create our clue phrase list. Some of the linguistic structures and words in this study are included within others, so we have chosen the most inclusive ones. For example, we chose *-Aydl* among (*eğer ... -sAyDI*), (*keşke ... -Aydl*), *-Aydl*. The distribution of clue phrases in classes is depicted in Table 4.4 along with the AMCD dataset. Since we mainly followed the creation process of the EN-ext version of the AMCD dataset, there is a similarity between their distribution and ours.

To the best of our knowledge, six CFD datasets have been published so far. Table 4.5 shows the comparison of these datasets and our dataset. These six datasets are in three different languages: English, German and Japanese. English is the most studied among these languages. Our dataset is the first-ever Turkish CFD dataset. To make our dataset compatible with the other CFD datasets, we mainly followed the creation process of AMCD EN-ext dataset. Our dataset has a similar positive class distribution to most of the datasets.

As a CFD dataset, our dataset contains sentences from forums, literary and news

Table 4.3. Turkish counterfactual dataset statistics

Dataset	Positive	Negative	Total	CF %
TRCD	640	4360	5000	12.8
TRCD w/ CP	615	1885	2500	24.6
TRCD w/o CP	25	2475	2500	1.0

which is the most diverse CFD dataset so far. Detailed domain distribution of our dataset in positive and negative classes are shown in Table 4.6. These domain categories are gathered from TNC. More details for our dataset related to clue phrases and random word distribution can be found in Appendix A.

Basic Query

Standard Query (Without PoS tag)
PoS Query (Lemma)
PoS Query (Affix)
Similarity Query

Window Span

-5

◀

▶

5

Window Span

Year

1 989

◀

▶

2 013

Year

Figure 4.1. Similarity query option in TNC

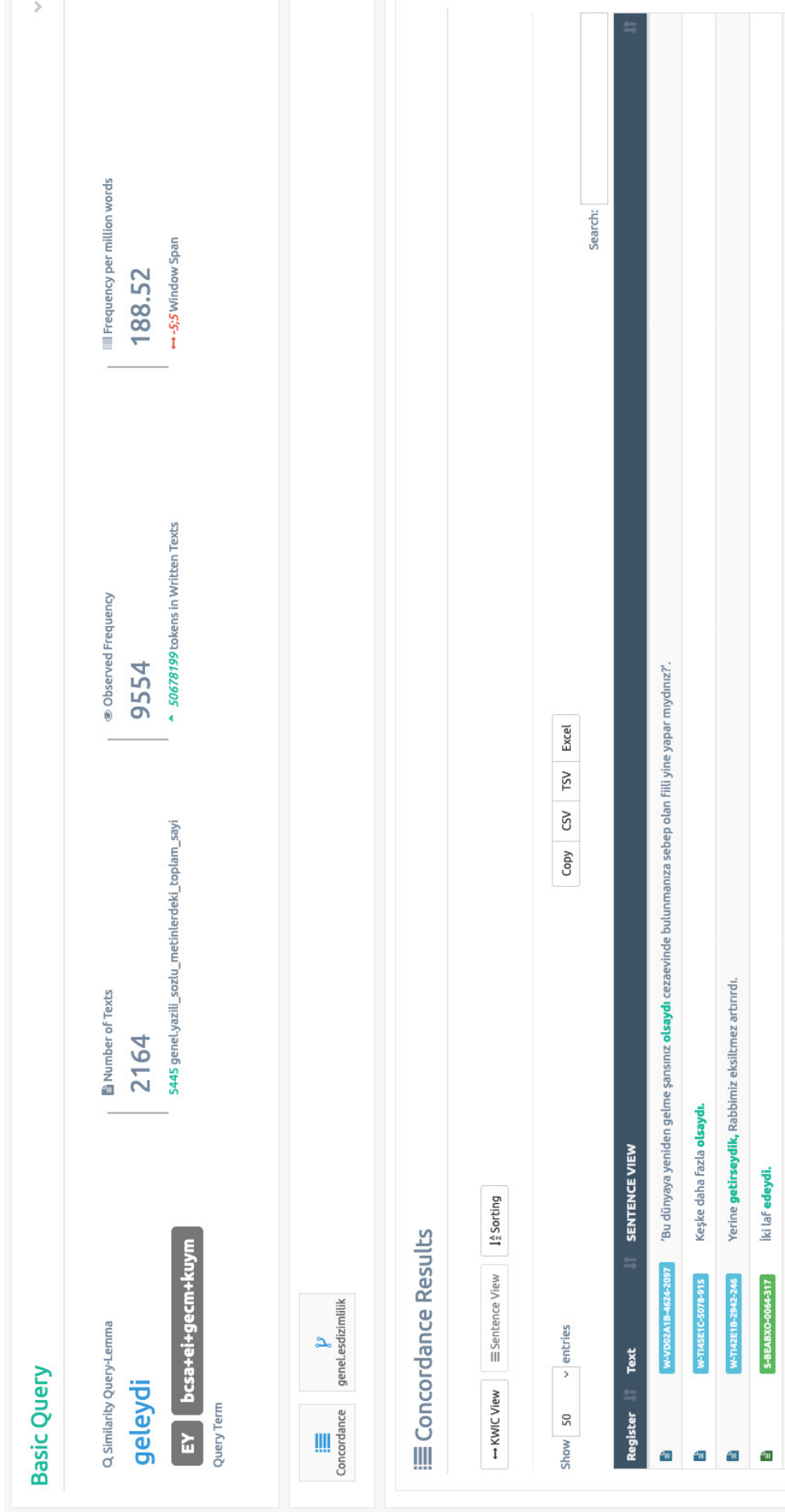


Figure 4.2. Similarity query result for *geleydi*

Basic Query

Standard Query (Without PoS tag)
PoS Query (Lemma)
PoS Query (Affix)
Similarity Query

Verb (VB) ▼

avsa+vi+past+pagr

Search

Window Span

-5

5

Window Span

Year

1 989

2 013

Year

Figure 4.3. PoS Affix query option in TNC

Basic Query

Q: PoS Query-Affix

VB

avsa+vi+past+pagr

Query Term

Number of Texts
2164
5445 genelyazili_sozlu_metinlerdeki_toplam_sayi

Observed Frequency
9556
50678199 tokens in Written Texts

Frequency per million words
188.56
-5.5 Window Span

Concordance

genel.esdizimlilik

Concordance Results

← KWIC View
≡ Sentence View
⚙ Sorting

Show 50
▼ entries

Copy
TSV
Excel

Register	Text	SENTENCE VIEW
E	W-1D37CZA-0304-1543	"Keşke, kalbinden veya bõbreğinden rahatsız olsaydı da, ameliyat olup kurtulsaydı.
E	W-UD2A1B-4618-1822	Hemcinsleri de kendisini ayıplar ve "bir neler çekiyorsuz, çocuklar için kattansaydı " derler.
E	W-0131CZA-1330-2254	Hayatını, hepimizin esasında birer kaybeden; kazanmanın bir illüzyon, bir yalan, bir banallik -HİÇLİK- esasında olduğu üstüne kursaydı? Hayatta kalmaz mıydı? Şimdi, buralarda olmaz mıydı?
E	W-DD26E1B-2846-1798	Şüphesiz girişim askeri açıdan "başarılı" sonuçansaydı , ANAP hükümeti erken seçimde kullanacağı önemli bir koz edinmiş olacaktır, ordu ise yıpranan prestijini onardığını düşünecek ve sorunun "böyle" halledileceğine dair inancını yeniden ve daha bir güvenle empoze etme imkânına sahip olacaktır.

Figure 4.4. PoS Affix query result for **VB + avsa+vi+past+pagr**

Basic Query

Standard Query (without PoS tag)	PoS Query (Lemma)	PoS Query (Affix)	Similarity Query
----------------------------------	-------------------	-------------------	------------------

özgür

genel.yazildigi_gibi

Search

Window Span

-5

5

Year

1989

1995

2001

2007

2013

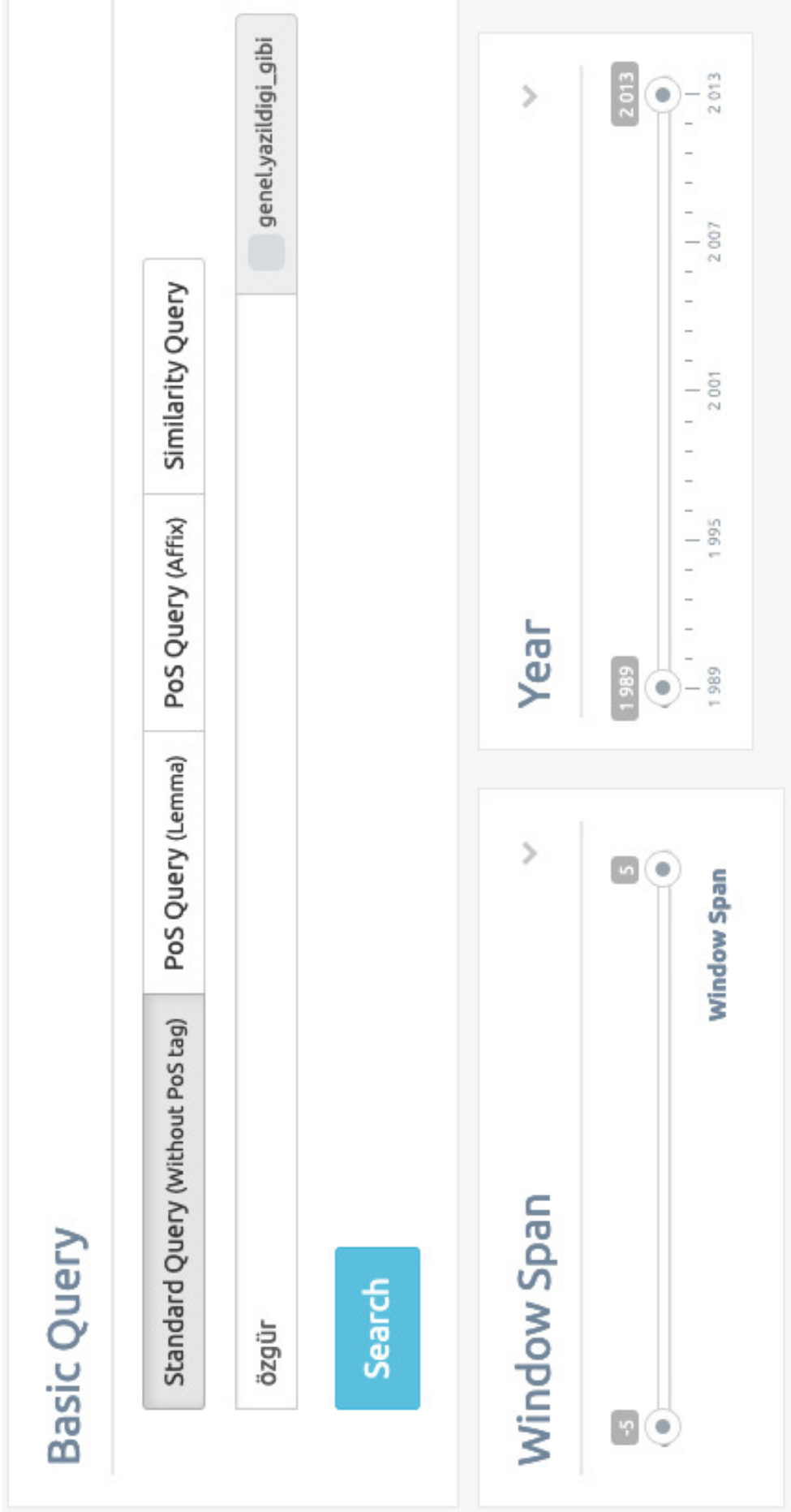
The image shows a web interface for a text analysis tool. At the top, there's a 'Basic Query' section with four tabs: 'Standard Query (without PoS tag)', 'PoS Query (Lemma)', 'PoS Query (Affix)', and 'Similarity Query'. The first tab is selected. Below the tabs is a text input field containing the word 'özgür'. To the right of the input field is a checkbox labeled 'genel.yazildigi_gibi'. Below the input field is a blue 'Search' button. Below the search section are two sliders. The first is labeled 'Window Span' and has a range from -5 to 5, with the current value set to 5. The second is labeled 'Year' and has a range from 1989 to 2013, with the current value set to 2013.

Figure 4.5. Standard Query option in TNC

Basic Query

Q. Standard Query
özgür
Query Term

Number of Texts
1232
5445 genel.yazili_sozlu_metinlerdeki_toplam_sayi

Observed Frequency
4452
50678199 tokens in Written Texts

Frequency per million words
87.85
-5.5 Window Span

Concordance

genel.esdzimlilik

Concordance Results

KWIC View

Sentence View

Sorting

Copy

CSV

TSV

Excel

Show 50 entries

Search:

Register	Text	SENTENCE VIEW
W-TC08A0A-0203-1344	yüzyilin başında, edebiyat taşıyan yazarların başında gelen Leylâ Erbil, yazarın bir kopyacıya, taklitçiye dönüştüğü, bağımsız bir birey olamayacağı, dolayısıyla "öçlüğü" savlanan koşullarda kaybolmamış, özgür zihni ve özgürleşme yazısıyla, itiraz eden, sorgulayan tavırla, etik terçhlerle dayanan estetiğiyle başlı başına bir odak olmuştur.	
W-ID02ZATB-4814-1864	Yine, özgür yurttaşlar bir başkasının hesabına çalışmayı ve ondan emir almayı onur kırıcı gördüklerinden, köleler ticaret ve bankacılık alanlarında muhasebe, temsilcilik ve yöneticilik gibi nitelikli işlerde çalışabiliyorlardı.	
W-V45FFD-7708-1888	A. Lider olmak için özgür olmak gerekir	
W-V414B1A-1605-1290	Anayasa Koyucu özgür ve demokratik bir düzeni kurmakla yurttaşların irade ve düşüncelerini özgür ve açık olarak kurma sürecini de öngörmüş olmaktadır.	

Figure 4.6. Standard Query result for word özgür

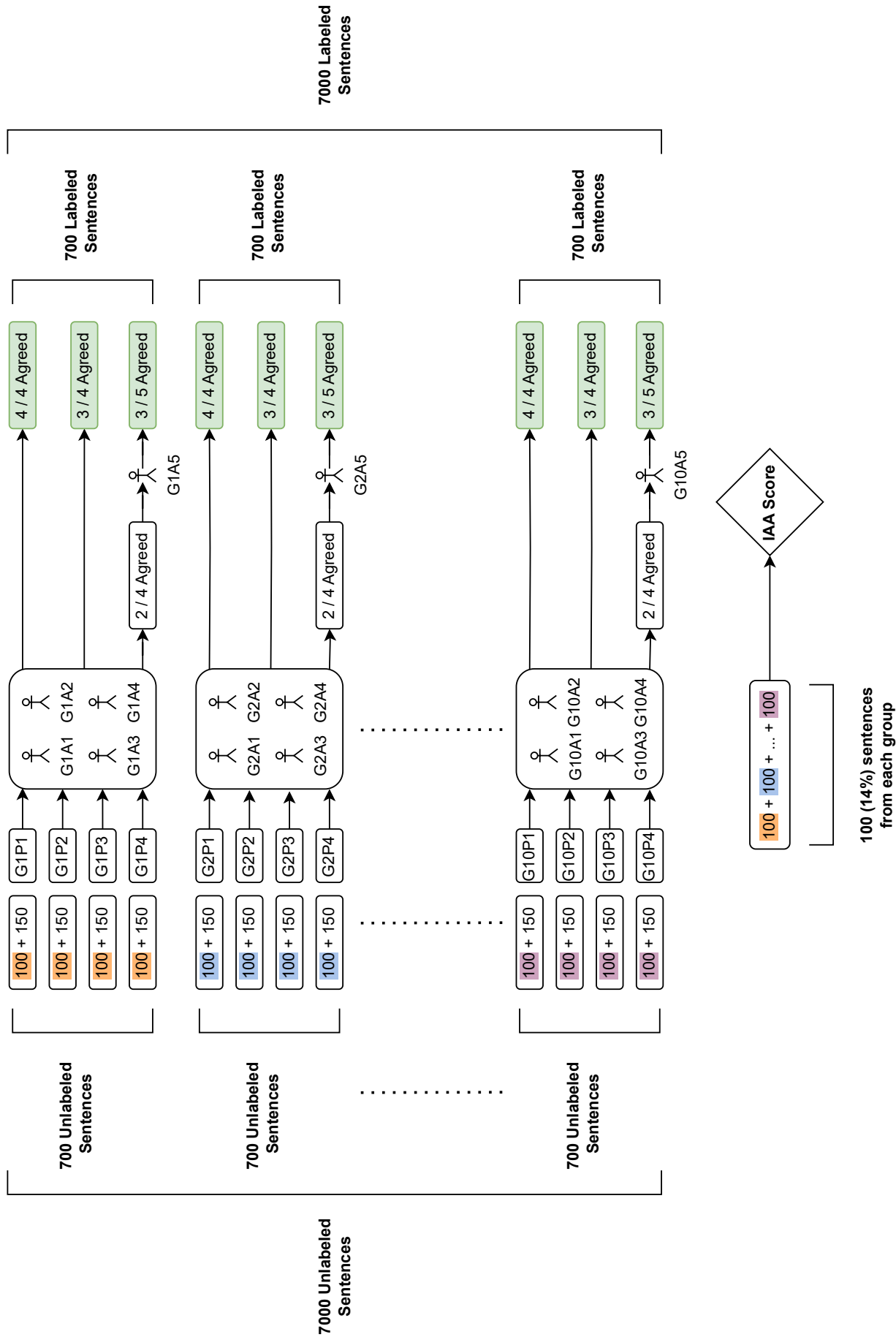


Figure 4.7. The annotation process

Table 4.4. CFD datasets clue phrase distribution comparison

Dataset	Language	N	CP % in Pos.	CP % in Neg.	CP % in Data	CF %
(O’Neill et al., 2021)	EN	29	100.	100.	100.	18.9
	EN-EXT	29	92.6	45.3	50.2	10.0
	DE	27	100.	100.	100.	69.1
	JA	70	100.	100.	100.	9.5
This work	TR	10	96.1	43.2	50.	12.8

Table 4.5. CFD datasets comparison

Dataset	Domain	Construction	Annotation	Language	Size	CF %
(Son et al., 2017)	Twitter	keywords filtering	manual (unknown)	English	1637	9.35
(Yang et al., 2020)	News: finance, politics, healthcare	keywords filtering, pattern matching	manual (crowdsourcing, strong agreement)	English	20000	11.0
(O’Neill et al., 2021)	Amazon Reviews	keywords filtering	manual (curated by linguists)	English	5023	18.9
				English (EXT)	10000	10.0
				German	7000	69.1
This work	News, Forums, Literary	keywords filtering	manual (crowdsourcing, substantial agreement)	Japanese	7000	9.5
				Turkish	5000	12.8

Table 4.6. Domain statistics of the Turkish CFD dataset

Domain	# in P.	# in N.	% in P.	% in N.	% in D.
Imaginative prose	245	1008	38.28	23.12	25.06
Informative: World affairs	98	811	15.31	18.6	18.18
Informative: Leisure	137	694	21.41	15.92	16.62
Informative: Social science	59	594	9.22	13.62	13.06
Informative: Commerce and finance	25	320	3.91	7.34	6.9
Informative: Arts	38	302	5.94	6.93	6.8
Informative: Applied science	12	278	1.88	6.38	5.8
Informative: Belief and thought	24	207	3.75	4.75	4.62
Informative: Natural and pure sciences	2	146	0.31	3.35	2.96
Total	640	4360	100.0	100.	100.

CHAPTER 5

EXPERIMENTS

In this section, we present the results of our experiments on various aspects of the CFD task. We mainly used The Matthews Correlation Coefficient (MCC) (Matthews, 1975) metric to evaluate our results since CFD datasets are imbalanced and MCC is a suitable metric for this (Chicco and Jurman, 2020). We also used the macro averaged F1 score to be consistent with the prior CFD works. Here are definitions of the MCC and F1 scores:

- **MCC:** The Matthews Correlation Coefficient (MCC) (Matthews, 1975) is another evaluation metric used in classification problems, especially when datasets are imbalanced. In the context of imbalanced classes, the MCC is a good metric that uses both underrepresented and overrepresented classes, unlike other metrics such as accuracy.

The MCC produces a high score only if the predictions obtain good results in all of the four confusion matrix categories (True positive, False positive, True negative, False negative) (Chicco and Jurman, 2020). The MCC’s range is from -1 to $+1$. A $+1$ represents a perfect prediction, 0 is a random prediction, and -1 indicates an inverse prediction.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (5.1)$$

- **F1 Score:** The F1 score is a measure of a model’s accuracy in classification problems. It is the harmonic mean of precision and recall. Precision is the number of correct positive predictions (True Positive) in proportion to the number of all positive predictions. Recall is the number of True Positive in proportion to all actual positive data points.

$$F1 = 2 * \frac{precision * recall}{precision + recall} \quad (5.2)$$

In the experiments, we used English (EN-ext), German (DE), and Japanese (JP) parts of the AMCD dataset, Semeval dataset and our dataset TRCD. We denoted the EN-ext part with AMCD-EN, the DE part with AMCD-DE and the JP part with AMCD-JP.

Lastly, detailed classification results of the TRCD dataset were presented in Subsection 5.5..

For the experiments, we fine-tuned a multilingual BERT (mBERT) ¹ and an XLM-RoBERTa (XLM-R)² model for all languages and BERTurk³ for only Turkish. To implement the models we used the Transformers⁴ library. The model is depicted in Figure 5.1. We fine-tuned the pre-trained model and the classifier layer in our experiments.

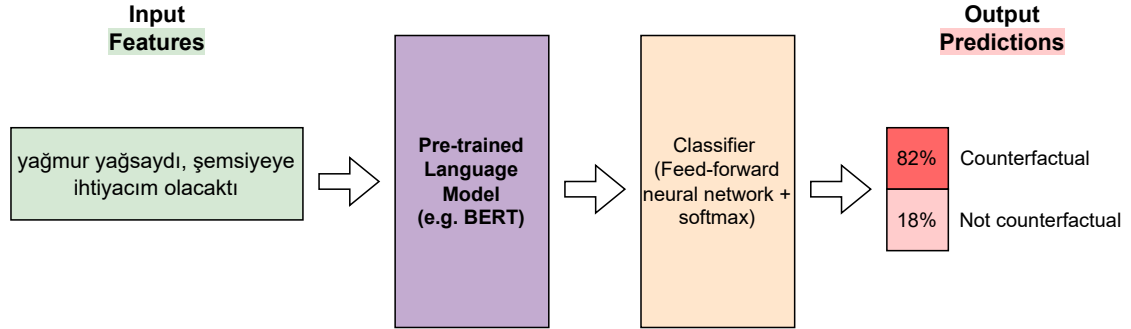


Figure 5.1. The classification model

Hyperparameters are tuned with 80% train and 20% test split of our TRCD dataset with 5-fold cross-validation. After the tuning, we used 0.00001 for the learning rate and 32 for the training batch size. We did not change the other parameters of the classifier and pre-trained models.⁵

Three experiments are conducted in this section: (1) the clue phrase effect on CFD task (subsection 5.1.), (2) cross-dataset performance of available CFD datasets (subsection 5.2.) and (3) combining CFD datasets for model fine-tuning (subsection 5.3.). In experiments 2 and 3 we obtained zero-shot cross-lingual classification results as well. The results of these zero-shot classification experiments will be discussed at the end of the section 5.3..

5.1. The Effect of Clue Phrases

In this experiment, we evaluated the effect of the clue phrases we used in the dataset creation process. (O’Neill et al., 2021) conducted this experiment on their dataset,

¹<https://huggingface.co/bert-base-multilingual-uncased>

²<https://huggingface.co/xlm-roberta-base>

³<https://huggingface.co/dbmdz/bert-base-turkish-cased>

⁴<https://github.com/huggingface/transformers>

⁵The fine-tuning script can be found at <https://github.com/dopc/turkish-counterfactual-recognition>

AMCD. To do that, they fine-tuned mBERT on AMCD with and without masking the clue phrases while the training. The results showed that the **no mask** (training without masking) option has slightly better performance than the **mask** (training with masking) fine-tuning option, but the performance difference was not significant. So, they showed that the model does not overfit the clue phrases and can classify sentences without seeing them.

We fine-tuned mBERT and XLM-R models for all the CFD datasets and BERTurk for the TRCD. The pad token of the fine-tuned model was used (i.e. *[PAD]* for BERT-based, *<pad>* for RoBERTa-based models) for masking the clue phrases. We replaced the clue phrase in a sentence with the pad token before feeding the sentence to the model. The tokenizers of these models can accurately encode the pad tokens existing in the raw text.

Table 5.1. MCC and F1 (macro) results for datasets with **mask** and **no mask** options

Train	Test	Masking	XLM-R		mBERT		BERTurk	
			MCC	F1	MCC	F1	MCC	F1
TRCD	TRCD	mask	0.0	46.6	11.0	48.9	53.3	74.0
		no mask	60.5	79.5	59.1	79.6	66.1	81.9
Semeval	Semeval	mask	79.2	89.5	69.1	84.1	-	-
		no mask	81.9	90.9	76.7	88.3	-	-
AMCD-JP	AMCD-JP	mask	0.0	47.3	0.0	47.3	-	-
		no mask	64.8	81.9	56.6	78.0	-	-
AMCD-DE	AMCD-DE	mask	80.4	90.1	39.2	58.1	-	-
		no mask	81.1	90.5	78.2	89.1	-	-
AMCD-EN	AMCD-EN	mask	73.2	86.6	77.3	88.6	-	-
		no mask	80.7	90.3	80.3	90.0	-	-

The results in Table 5.1 show that with our fine-tuning and masking method, the multilingual models had their lowest MCC score with **mask** option. Moreover, zero MCC scores for Turkish and Japanese could be caused by tokenization issues and coverage of these languages. Both Turkish and Japanese clue phrases are more difficult to handle by tokenizers since they are part of verb/adjective inflections (O’Neill et al., 2021). We also fine-tuned a Turkish-specific BERT⁶ (BERTurk) model (Schweter, 2020), and it did not get a significant MCC score drop. This suggests that the BERTurk model could learn the CFD task better for Turkish compared to the other models. The reason could be that the BERTurk model tokenizes Turkish better with its Turkish-specific tokenizer.

XLM-R did not get any significant performance drop for AMCD-EN or AMCD-DE. However, the mBERT experienced a half MCC score drop for AMCD-DE with **mask**.

⁶<https://huggingface.co/dbmdz/bert-base-turkish-cased>

This performance drop might be the result of an over-fitting issue caused by the higher positive example percentage of the AMCD-DE dataset (69.1%) compared to the other datasets (around 12%).

5.2. Cross-Dataset Performance

Developing a CFD dataset for a new language is not an easy task. It involves clue phrase list creation and manual annotation of sentences. (O’Neill et al., 2021) experimented with Machine Translation (MT) for cross-lingual CFD classification. They fine-tuned the mBERT model with the AMCD-EN dataset and used this model on translated versions of AMCD-DE and AMCD-JP datasets. However, they got poor performance in this setup.

Table 5.2. Cross-dataset Performance of the models with **no mask** option

Train	Test	XLM-R		mBERT	
		MCC	F1	MCC	F1
AMCD-DE		81.1	90.5	78.2	89.1
AMCD-EN		23.5	39.4	2.2	23.5
AMCD-JP	AMCD-DE	9.3	26.4	-0.5	23.9
Semeval		42.8	60.5	9.5	26.5
TRCD		34.6	51.5	10.5	27.9
AMCD-DE		57.3	74.8	32.8	64.2
AMCD-EN		80.7	90.3	80.3	90.0
AMCD-JP	AMCD-EN	22.8	52.8	0.0	47.3
Semeval		73.5	86.7	69.1	83.5
TRCD		45.2	68.2	5.8	50.6
AMCD-DE		56.3	78.0	-2.6	47.6
AMCD-EN		14.0	50.3	0.0	47.3
AMCD-JP	AMCD-JP	64.8	81.9	56.6	78.0
Semeval		53.0	75.5	0.0	47.3
TRCD		29.1	57.6	0.0	47.3
AMCD-DE		21.6	38.2	20.0	48.0
AMCD-EN		53.3	73.6	28.7	50.7
AMCD-JP	Semeval	7.8	47.9	-0.8	47.3
Semeval		81.9	90.9	76.7	88.3
TRCD		53.7	76.6	8.7	49.5
AMCD-DE		45.7	69.3	22.6	60.2
AMCD-EN		25.2	53.6	0.0	46.6
AMCD-JP	TRCD	37.3	60.6	0.0	46.6
Semeval		57.9	78.7	0.0	46.6
TRCD		60.5	79.5	59.1	79.6

We experimented with cross-lingual CFD classification without MT. We fine-tuned **no mask** the mBERT and XLM-R models with AMCD, Semeval and TRCD datasets separately and tested these models on each test set. The results in Table 5.2 show that the XLM-R model is much better for cross-lingual classification compared to the mBERT model since XLM-R is specially developed for cross-lingual tasks. In the rest of the section, we only discussed XLM-R results.

We got the best results for all test sets with the XLM-R model for the test set’s corresponding train set.

The XLM-R model, which is trained with the Semeval dataset, performed well for AMCD-JP and TRCD test sets. Interestingly, the AMCD-EN-trained model got performance drops on AMCD-JP and TRCD test sets compared to their own train sets’ performance. The apparent reason could be the differences in domain diversity between the Semeval and AMCD-EN datasets, as stated in (O’Neill et al., 2021). Both of these English-trained models saw a drop in performance on the AMCD-DE test sets. The results indicate that a domain-diverse CFD dataset (e.g. Semeval) could be a good candidate for cross-domain and cross-lingual CFD classification for especially relatively low-resourced languages, Japanese and Turkish in this case.

The XLM-R model trained with the AMCD-DE dataset outperformed the models trained with AMCD-EN on the AMCD-JP and TRCD datasets. This outcome suggests that the model trained with AMCD-EN may be overfitting its training data, as it also showed poor performance on the AMCD-DE dataset.

5.3. Combined Datasets Performance

In this section, we experimented with combining training and validation data of CFD datasets for fine-tuning the multilingual models XLM-R and mBERT. (O’Neill et al., 2021) conducted a similar experiment with the mBERT model to show the compatibility of their dataset with the Semeval dataset (Yang et al., 2020). In their experiment, both of the datasets were in English. Our experiment combines the datasets in a more comprehensive way by a cross-lingual setup, and we also showed the effect of masking the clue phrases.

The XLM-R and mBERT models are fine-tuned with three different training and validation dataset combinations. Below are these fine-tuning options:

- **(a)** using only the training counterpart of the test dataset (e.g. fine-tuning with TRCD for TRCD testing)
- **(b)** combining all the datasets except the test dataset (e.g. fine-tuning with AMCD-DE, AMCD-EN, AMCD-JP and Semeval for TRCD testing)

- **(c)** combining all training datasets (i.e. fine tuning with AMCD-DE, AMCD-EN, AMCD-JP, Semeval, TRCD for all testings)

Table 5.3. Results of combining dataset experiment. We used abbreviations for the datasets: E for AMCD-EN, D for AMCD-DE, J for AMCD-JP, S for Semeval and T for TRCD

Train	Test	Masking	XLM-R		mBERT	
			MCC	F1	MCC	F1
AMCD-DE	AMCD-DE	mask	80.4	90.1	39.2	58.1
AMCD-DE		no mask	81.1	90.5	78.2	89.1
E + J + S + T		mask	44.7	62.1	11.6	28.7
E + J + S + T		no mask	40.3	58.1	20.8	41.1
E + D + J + S + T		mask	67.6	83.7	54.6	76.1
E + D + J + S + T		no mask	80.7	90.2	76.7	88.3
AMCD-EN	AMCD-EN	mask	73.2	86.6	77.3	88.6
AMCD-EN		no mask	80.7	90.3	80.3	90.0
D + J + S + T		mask	65.8	82.9	34.1	64.5
D + J + S + T		no mask	75.1	87.5	65.8	82.3
E + D + J + S + T		mask	70.7	85.3	49.3	70.2
E + D + J + S + T		no mask	75.2	87.6	80.6	90.1
AMCD-JP	AMCD-JP	mask	0.0	47.3	0.0	47.3
AMCD-JP		no mask	64.8	81.9	56.6	78.0
E + D + S + T		mask	51.3	72.8	0.0	47.3
E + D + S + T		no mask	51.1	74.1	0.0	47.3
E + D + J + S + T		mask	49.6	72.9	0.0	47.3
E + D + J + S + T		no mask	66.0	82.8	29.1	59.6
Semeval	Semeval	mask	79.2	89.5	69.1	84.1
Semeval		no mask	81.9	90.9	76.7	88.3
E + D + J + T		mask	55.9	75.8	33.3	62.3
E + D + J + T		no mask	43.8	64.4	35.3	57.9
E + D + J + S + T		mask	77.8	88.6	57.3	76.2
E + D + J + S + T		no mask	80.9	90.4	72.6	85.8
TRCD	TRCD	mask	0.0	46.6	11.0	48.9
TRCD		no mask	60.5	79.5	59.1	79.6
E + D + J + S		mask	50.1	74.4	5.9	48.7
E + D + J + S		no mask	49.0	73.8	21.8	57.7
E + D + J + S + T		mask	53.0	75.7	0.0	46.6
E + D + J + S + T		no mask	56.7	77.3	30.7	57.6

In the training option **b** and **c**, we sampled the training and validation part of the combined dataset as the size of the training and validation counterpart of the test dataset. For example, for testing the TRCD dataset, we got a 4000-sized sample from the combined training dataset since the TRCD train dataset has 4000 sentences. With this sampling, we

aimed not to introduce any bias which is caused by the training and validation dataset size differences. In each sample, we used the same seed value.

In Table 5.3, we abbreviated the names of combined datasets for better visualization: **D** for AMCD-DE, **J** for AMCD-JP, **E** for AMCD-EN, **S** for Semeval, and **T** for the TRCD dataset.

The AMCD-JP test set achieved the best results with the XLM-R model when fine-tuned using option **c**. The other four test datasets got the best results with the fine-tuning option **a** with the XLM-R model.

Combining datasets for fine-tuning with the XLM-R model, in **b** and **c**, generally makes the models more robust for masking the clue phrases. With the option **b**, we obtained the most robust models for masking the clue phrases with the XLM-R model, except for AMCD-EN. Moreover, fine-tuning option **b** consistently achieved a higher MCC score with the **mask** option, compared to the **no mask** option. This pattern held for all datasets tested with the XLM-R model, except for AMCD-EN.

5.4. Zero-shot Classification Results

The last two experiments utilized zero-shot cross-lingual setups. In the subsection 5.2., we used dataset pairs (like Semeval - TRCD) with differing train and test parts. In the subsection 5.3., we employed the fine-tuning option **b**.

Table 5.4. Zero-shot classification macro-F1 scores on the 5 CFD datasets

Method	AMCD-DE	AMCD-EN	AMCD-JP	Semeval	TRCD
Our best supervised result	90.5	90.3	89.3	90.9	81.9
Ushio and Bollegala (2022)	73.0	-	82.9	-	-
Our best zero-shot result	62.1	75.1	78.0	75.8	78.7

To the best of our knowledge, (Ushio and Bollegala, 2022) achieved the state-of-the-art (SotA) result for zero-shot cross-lingual classification results for the AMCD-DE and AMCD-JP datasets. As shown in Table 5.4, we achieved a 78.0 macro F1 score for AMCD-JP, reaching 94% of the SotA performance by fine-tuning the XLM-R model with the AMCD-DE dataset with **no mask**. For the AMCD-DE dataset, the XLM-R model, which is fine-tuned by the combined AMCD-EN, AMCD-JP, Semeval and TRCD, achieved a 62.1 macro F1 score (85% of the SotA result) with **mask** option.

On the other hand, for the TRCD dataset, we obtained the highest MCC score of 66.1 (81.9 macro F1) with the BERTurk model with **no mask**. The XLM-R model with **no**

mask which is fine-tuned with the Semeval model achieved the best zero-shot cross-lingual result with a 57.9 MCC (78.7 F1) score which is 88% of the BERTurk performance.

These results demonstrate that cross-paired and combined CFD datasets are promising candidates for fine-tuning classifier models in languages that have not been previously studied within the context of the CFD task.

The results also suggest that our dataset, TRCD, is compatible with other CFD datasets for cross-dataset fine-tuning and dataset combining options.

5.5. TRCD Detailed Classification Results

In this section, we present a more detailed classification of results for our TRCD dataset. We used two different fine-tuned models’ results: **(a)** the BERTurk model which is fine-tuned with TRCD and **no mask** option. This model achieved the highest MCC score for the TRCD dataset. **(b)** the XLM-R model which is fine-tuned with **no mask** option and all the datasets but TRCD, the best zero-shot cross-lingual model for the TRCD dataset.

In Table 5.5, the confusion matrix of the model **a** is depicted. The model got almost the same results for positive class instances (**CF**) with **mask** and **no mask** options. However, the **mask** option led to significant misclassification of negative class instances (**Not CF**). This result suggests that the model **a** might be over-fitted with clue phrases for classifying counterfactuality.

On the other hand, the model **b** performed worse on negative and better on positive class instances with **mask** as shown in Table 5.6. This outcome suggests that the model **b** assigns less importance to clue phrases during classification. This could also be a consequence of a higher percentage of positive class instances in the training dataset for the model **b** compared to the model **a**.

Table 5.5. Confusion Matrix of TRCD fine-tuned BERTurk model with **no mask** (Predicted_{nm}) and **mask** options (Predicted_m)

		Predicted _{nm}		Predicted _m	
		CF	Not CF	CF	Not CF
Actual	CF	560	7	561	6
	Not CF	37	46	51	32

In Table 5.7, we have presented the macro F1 scores per class for both models **a**

Table 5.6. Confusion Matrix of (E + D + J + S) fine-tuned XLM-R model with **no mask** (Predicted_{nm}) and **mask** options (Predicted_m)

		Predicted _{nm}		Predicted _m	
		CF	Not CF	CF	Not CF
Actual	CF	548	19	506	61
	Not CF	46	37	26	57

and **b** for each Turkish clue phrase. Both **a** and **b** models classified pure (all positive class or all negatives class) and almost pure clue phrases (e.g. *-sA*) fully correctly.

For the negative class instances, the model **a** is generally not affected by masking the clue phrases. The model **b** made more wrong classifications with **mask** on negative class instances for almost all clue phrases.

For the positive class, on the other hand, the model **a** made worse classifications with **mask** on the positive class compared to the **no mask** option. And the model **b** is the opposite.

Both models made an exception for the clue phrases *-ArdI* and *-mAllydI*. For example, the **mask** option makes the model **a** better and the model **b** worse on positive class instances with *-mallydI* clue phrase. As shown in Table 4.1, the clue phrase *-mallydI* has the highest percentage of **undecided** annotations. Another clue phrase *-AbilArdI* which has a significantly high **undecided** annotation rate classified poorly by both models. That result shows us that clue phrases which are challenging to annotate by human annotators are also challenging to learn by both models.

We also showed the classification results of both models for each domain of our dataset in Table 5.8. The model **a** had the same pattern with the classification for each clue phrase. It again demonstrated similar performance for the negative class and worse for the positive class with **mask** compared to **no mask**.

However, the model **b** slightly deviated from its previous trend which is in the classification of each clue phrase. It achieved similar classification performance with **mask** compared to **no mask** for negative class instances. For the positive class, it generally achieved better classification with **mask**, with a few exceptions (i.e. *Informative: Leisure*, *Informative: Arts*, *Informative: Belief and thought*). We previously discussed a similar pattern, referring to the **undecided** annotation distribution among clue phrases, in the above classification results. However, the distribution of **undecided** sentences across the domain classes is uniform. We suggest this issue as a potential topic for future research.

Table 5.7. Classification results for the BERTurk model and Zero-shot model for each clue phrase of the TRCD dataset. **Count** refers to the sentences with the clue phrase in the TRCD dataset. **Pos. %** refers to the positive percentage of the sentences with the clue phrase. In the scores, **N** and **P** refer to negative and positive class F1 scores, respectively.

Clue Phrase	Count	Pos.%	BERTurk		Zero-shot	
			F1 _{nm}	F1 _m	F1 _{nm}	F1 _m
-sA	65	1.5	N: 100.0 P: 100.0	N: 100.0 P: 100.0	N: 100.0 P: 100.0	N: 100.0 P: 100.0
-AmAzdI	55	7.3	N: 95.2 P: 0.0	N: 94.2 P: 0.0	N: 94.2 P: 0.0	N: 79.1 P: 25.0
-ArdI	48	14.6	N: 92.0 P: 22.2	N: 93.2 P: 25.0	N: 93.2 P: 25.0	N: 81.0 P: 11.8
-mAlIydI	42	38.1	N: 83.9 P: 54.5	N: 78.8 P: 22.2	N: 52.0 P: 29.4	N: 58.3 P: 44.4
-mAzDI	36	16.7	N: 89.2 P: 0.0	N: 89.2 P: 0.0	N: 90.9 P: 0.0	N: 82.1 P: 37.5
-AbilArdI	30	23.3	N: 86.3 P: 22.2	N: 84.6 P: 0.0	N: 84.0 P: 20.0	N: 64.9 P: 43.5
-AydI	29	93.1	N: 0.0 P: 94.5	N: 18.2 P: 80.9	N: 28.6 P: 90.2	N: 0.0 P: 96.4
-sAlArDI	10	90.0	N: 0.0 P: 94.7	N: 0.0 P: 82.4	N: 0.0 P: 57.1	N: 0.0 P: 82.4
-AcAkDI	2	0.0	N: 100.0 P: 0.0	N: 100.0 P: 0.0	N: 100.0 P: 0.0	N: 100.0 P: 0.0
-AymIş	2	100.0	N: 0.0 P: 100.0	N: 0.0 P: 100.0	N: 0.0 P: 100.0	N: 0.0 P: 100.0

Table 5.8. Classification results for the BERTurk model and Zero-shot model on each domain of the TRCD dataset. **Count** refers to the sentences with the domain in the TRCD dataset. **Pos. %** refers to the positive percentage of the sentences with the domain. In the scores, **N** and **P** refer to negative and positive class F1 scores respectively.

Domain	Count	Pos.%	BERTurk		Zero-shot	
			F1 _{nm}	F1 _m	F1 _{nm}	F1 _m
Imaginative prose	177	23.2	N: 91.1 P: 58.1	N: 89.6 P: 45.6	N: 86.3 P: 43.5	N: 81.1 P: 52.0
Informative: World affairs	125	15.2	N: 96.7 P: 80.0	N: 95.0 P: 64.5	N: 94.5 P: 60.0	N: 93.7 P: 69.8
Informative: Leisure	91	8.8	N: 98.2 P: 80.0	N: 97.1 P: 54.5	N: 98.2 P: 80.0	N: 95.7 P: 66.7
Informative: Social science	85	7.1	N: 98.1 P: 66.7	N: 97.5 P: 50.0	N: 96.9 P: 44.4	N: 95.5 P: 46.2
Informative: Commerce and finance	42	7.1	N: 95.0 P: 0.0	N: 95.0 P: 0.0	N: 95.0 P: 0.0	N: 93.7 P: 0.0
Informative: Applied science	38	2.6	N: 98.7 P: 0.0	N: 98.7 P: 0.0	N: 98.7 P: 0.0	N: 97.2 P: 50.0
Informative: Arts	38	5.3	N: 100.0 P: 100.0	N: 100.0 P: 100.0	N: 100.0 P: 100.0	N: 95.7 P: 57.1
Informative: Belief and thought	31	9.7	N: 100.0 P: 100.0	N: 98.2 P: 80.0	N: 98.2 P: 85.7	N: 96.3 P: 75.0
Informative: Natural and pure sciences	23	0.0	N: 100.0 P: 0.0	N: 100.0 P: 0.0	N: 100.0 P: 0.0	N: 100.0 P: 0.0

CHAPTER 6

CONCLUSION

In this work, we released the first-ever CFD dataset for Turkish (TRCD) and the result of the classification results of the classification methods we have tried.

In our work, we used a Turkish-specific counterfactual definition proposed by (Üzüm, 2020) and utilized a clue phrase list from the linguistic structures in this study. These clue phrases are used for the data collection process in the TNC query interface to collect sentences with clue phrases. We also collect sentences without clue phrases to avoid introducing a selection bias due to the clue phrases. Then, we trained human annotators using the guidelines we had prepared. As a result, we obtained 5000 labelled sentences, 14% of which were annotated by four different annotators, achieving a 65.04% inter-annotator agreement score. We have encountered challenges since the meanings of the Turkish clue phrases are highly context-dependent, which makes developing a clue phrase list for the Turkish a challenging problem. This also introduces challenges for annotating the Turkish sentences for the CFD task. Furthermore, the clue phrases are part of verb inflections.

For classification, we utilized two multilingual models (mBERT and XLM-R) and one Turkish-specific model (BERTurk), all fine-tuned. BERTurk achieves the best classification performance for the TRCD. We tested for potential selection bias by masking the clue phrases. We also experimented with cross-dataset and dataset-combined fine-tuned models for classification. In these dataset adaptation experiments, we reported that our dataset TRCD is compatible with the other CFD datasets. We also proposed a novel zero-shot cross-lingual classification method for the CFD task by combining the CFD datasets. Our methods achieved 94% and 85% of SotA zero-shot cross-lingual performance (Ushio and Bollegala, 2022) for Japanese and German for the CFD task, respectively. However, we could not compare our zero-shot result for Turkish since we could not reproduce the (Ushio and Bollegala, 2022) work.

Lastly, we presented detailed classification results for each clue phrase and text domain in our dataset. We also shared qualitative analysis of different classification methods for our dataset.

6.1. Limitations

A recent work (Denizer, 2023) studied counterfactuals for the Turkish languages in a more comprehensive way compared to the work we used (Üzüm, 2020). After this work, we have not changed our counterfactual definition and clue phrases, as these two studies agree substantially on the counterfactual definition and the Turkish clue phrases.

6.2. Future Work

With the rise of zero-shot cross-lingual classifying methods, developing a dataset for a language which is not studied in a task has seen reduced interest. (Ushio and Bollegala, 2022) achieved the SotA results for zero-shot cross-lingual CFD classification for German and Japanese CFD datasets. However, we had not been able to reproduce their work and test it on our dataset. If we could do that, we could see its performance on our dataset and better understand the value of developing a CFD dataset for Turkish. Therefore, we will reproduce this work and apply the methodology to our data.

REFERENCES

- Aksan, M., A. Koltuksuz, T. Sezer, Ü. Mersinli, et al. (2012). Construction of the turkish national corpus (tnc).
- Bird, S., E. Klein, and E. Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc."
- Chicco, D. and G. Jurman (2020, jan). The advantages of the matthews correlation coefficient (MCC) over f1 score and accuracy in binary classification evaluation. *BMC Genomics* 21(1).
- Çöltekin, Ç. (2010, May). A freely available morphological analyzer for Turkish. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).
- Denizer, F. U. (2023). *Counterfactuality in Turkish*. Ph. D. thesis.
- Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova (2019, June). 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)*, Minneapolis, Minnesota, pp. 4171–4186. Association for Computational Linguistics.
- Ding, X., D. Hao, Y. Zhang, K. Liao, Z. Li, B. Qin, and T. Liu (2020, December). HIT-SCIR 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*, Barcelona (online), pp. 354–360. International Committee for Computational Linguistics.
- Fajcik, M., J. Jon, M. Docekal, and P. Smrz (2020, December). 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*, Barcelona (online), pp. 437–444. International Committee for Computational Linguistics.
- Fleiss, J. (1971, November). Measuring nominal scale agreement among many raters. *Psychological bulletin* 76(5), 378—382.
- Hume, D. and P. F. Millican (2007). *An enquiry concerning human understanding*. Oxford world's classics (Oxford University Press). Oxford: Oxford University Press.

- Janocko, A., A. Larche, J. Raso, and K. Zembrroski (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.
- Liu, Y., M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR abs/1907.11692*.
- Lu, Y., A. Li, H. Lin, X. Han, and L. Sun (2020, December). ISCAS at SemEval-2020 task 5: Pre-trained transformers for counterfactual statement modeling. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, Barcelona (online), pp. 658–663. International Committee for Computational Linguistics.
- Matthews, B. (1975). Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure* 405(2), 442–451.
- Mikolov, T., I. Sutskever, K. Chen, G. Corrado, and J. Dean (2013a). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13*, Red Hook, NY, USA, pp. 3111–3119. Curran Associates Inc.
- Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean (2013b). Distributed representations of words and phrases and their compositionality. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger (Eds.), *Advances in Neural Information Processing Systems*, Volume 26. Curran Associates, Inc.
- Milmed, B. K. (1957). Counterfactual statements and logical modality. *Mind* 66(264), 453–470.
- Ojha, A. A., R. Garg, S. Gupta, and A. Modi (2020, December). IITK-RSA at SemEval-2020 task 5: Detecting counterfactuals. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, Barcelona (online), pp. 458–467. International Committee for Computational Linguistics.
- O’Neill, J., P. Rozenshtein, R. Kiryo, M. Kubota, and D. Bollegala (2021, November). I wish I would have loved this one, but I didn’t – a multilingual dataset for counterfactual detection in product review. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic, pp. 7092–7108. Association for Computational Linguistics.
- Ortiz Suárez, P. J., B. Sagot, and L. Romary (2019). Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. *Proceedings of the Workshop*

- on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, Mannheim, pp. 9 – 16. Leibniz-Institut für Deutsche Sprache.
- Pearl, J. and D. Mackenzie (2018). *The Book of Why: The New Science of Cause and Effect* (1st ed.). USA: Basic Books, Inc.
- Schweter, S. (2020, April). Berturk - bert models for turkish.
- Son, Y., A. Buffone, J. Raso, A. Larche, A. Janocko, K. Zembroski, H. A. Schwartz, and L. Ungar (2017, July). Recognizing counterfactual thinking in social media texts. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vancouver, Canada, pp. 654–658. Association for Computational Linguistics.
- Speer, R. (2022, September). rspeer/wordfreq: v3.0.
- Ushio, A. and D. Bollegala (2022, December). Zero-shot cross-lingual counterfactual detection via automatic extraction and prediction of clue phrases. In *Proceedings of the The 2nd Workshop on Multi-lingual Representation Learning (MRL)*, Abu Dhabi, United Arab Emirates (Hybrid), pp. 28–37. Association for Computational Linguistics.
- Üzüm, M. (2020). Türkçede karşıolgusallık: Korpus temelli bir inceleme. *Dilbilimde Güncel Tartışmalar*, 123–130.
- Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, Volume 30. Curran Associates, Inc.
- Yabloko, L. (2020, December). 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*, Barcelona (online), pp. 645–652. International Committee for Computational Linguistics.
- Yang, X., S. Obadinma, H. Zhao, Q. Zhang, S. Matwin, and X. Zhu (2020, December). SemEval-2020 task 5: Counterfactual recognition. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, Barcelona (online), pp. 322–335. International Committee for Computational Linguistics.

APPENDIX A

DETAILED TRCD DATASET STATISTICS

Table A.1. Clue phrases statistics of the Turkish CFD dataset

Clue phrase	# in P.	# in N.	% in P.	% in N.	% in D.
-sA	22	571	3.44	13.1	11.86
-AmAzDI	28	330	4.38	7.57	7.16
-ArdI	32	321	5.0	7.36	7.06
-mAzDI	42	254	6.56	5.83	5.92
-mAlIydI	112	173	17.5	3.97	5.7
-AbilArdI	83	192	12.97	4.4	5.5
-AydI	194	23	30.31	0.53	4.34
-sAlArDI	83	6	12.97	0.14	1.78
-AymIş	16	2	2.5	0.05	0.36
-AcAkDI	3	13	0.47	0.3	0.32
Total	615	1885	96.1	43.2	50.0

Table A.2. Random words statistics of the Turkish CFD dataset

Clue phrase	# in P.	# in N.	% in P.	% in N.	% in D.
halkın	0	204	0.0	4.68	4.08
uyum	1	202	0.16	4.63	4.06
yaklaşık	1	198	0.16	4.54	3.98
zamanda	0	198	0.0	4.54	3.96
için	4	186	0.62	4.27	3.8
tür	2	184	0.31	4.22	3.72
çıkan	3	171	0.47	3.92	3.48
kadar	0	172	0.0	3.94	3.44
söz	2	159	0.31	3.65	3.22
halde	6	151	0.94	3.46	3.14
özgür	1	144	0.16	3.3	2.9
belirlenmiştir	1	131	0.16	3.0	2.64
yok	1	121	0.16	2.78	2.44
liderleri	0	47	0.0	1.08	0.94
koşar	0	42	0.0	0.96	0.84
faydası	0	40	0.0	0.92	0.8
kanseri	3	32	0.47	0.73	0.7
toplar	0	25	0.0	0.57	0.5
aradığım	0	16	0.0	0.37	0.32
kanama	0	14	0.0	0.32	0.28
golf	0	13	0.0	0.3	0.26
sıralamada	0	9	0.0	0.21	0.18
varis	0	6	0.0	0.14	0.12
şirketiyle	0	5	0.0	0.11	0.1
alçakça	0	5	0.0	0.11	0.1
Total	25	2475	3.92	56.75	50.

APPENDIX B

QUALITATIVE ANALYSIS OF THE CLASSIFICATION OF THE TRCD DATASET

Table B.1. Qualitative Analysis for misclassified Positive class instances. The model which miss-classified the sentence given in the Models column. T: BERTurk, Z: the zero-shot model which fine-tuned XLM-R with (D + E + J + S) dataset combination, X: XLM-R model which fine-tuned with TRCD. The subscripts indicate masking strategy, *nm* for no masking and *m* for masking the clue phrases.

Counterfactual Sentence	Models
Çadır değildi ki sır saklayaydı.	T _{nm} T _m Z _{nm} X _{nm} X _m
"Kusur bende oldu, size danışmalıyım.	T _{nm} T _m Z _{nm} X _m
Bir taş attı diye elli yıl hapse atsaydın.	Z _{nm} X _m
Sol mememin yerine rahmimi alsalardı örneğin.	Z _{nm} X _m
MELTEM: Bu kadar kirlenebileceğim düşünemezdim!	T _{nm} T _m Z _{nm} X _{nm} X _m
Avukat: "Tamam, hatalıyım, bu ihtimalden bahsetmeliydim.	Z _{nm} X _m
Bilmek istemezdim çünkü saçlarım bu aralar çok dökülüyor.	T _{nm} T _m Z _{nm} Z _m X _{nm} X _m
İsterdim ki sesin sağır etsin kenti; isimle esasın sınırları.	T _{nm} Z _{nm} Z _m X _m
Bu duygudan kurtulmaya çalışmalıydı, ama elinde olmadığını sezinledi.	T _{nm} T _m Z _{nm} Z _m X _{nm} X _m
Kim görebilirdi ki yirmibeş sene sonra büyüüp yükseklerde uçacağımızı...	T _{nm} T _m Z _{nm} X _{nm} X _m
Onun için çekmeliydi acıyı ve sonra gözüne sürmeyi çekmeliydi altın gencin.	T _{nm} T _m Z _{nm} Z _m X _{nm} X _m
Saçları sarı, gözleri de mavi olsun çok isterdi ama bir çok istediği gibi bu da olamıyor.	T _{nm} T _m Z _{nm} Z _m X _{nm} X _m

Table B.2. Qualitative Analysis for misclassified negative class instances. The model which miss-classified the sentence given in the Models column. T: BERTurk, Z: the zero-shot model which fine-tuned XLM-R with (D + E + J + S) dataset combination, X: XLM-R model which fine-tuned with TRCD. The subscripts indicate masking strategy, *nm* for no masking and *m* for masking the clue phrases.

Non-counterfactual Sentence	Models
Tabii ki Özgür'ün önünde konuşamazdım.	T _m Z _m
Osmanlı'dan kalma kilitler olmalıydı bunlar.	Z _{nm} Z _m
Hayır, bu yüz, bu saçlar kendisinin olamazdı.	T _m Z _{nm} Z _m
Merdivenden aşağı inişleri, bir "dünya rekoru" kırmış olmalıydı.	Z _{nm} Z _m
Açık söyleyeyim, o an yöneticilerimizin yerinde olmak istemezdim.	T _{nm} T _m X _{nm}
Oysa bedenden kopamamış bir irade ne kadar özgür olabilirdi ki...	Z _m X _{nm}
Sonuçta bu harika oldu" diyen Cooper, "Daha iyi bir paskalya hediyesi olamazdı" diye ekledi.	T _{nm} Z _{nm} Z _m
Çiller'in ortaya koyduğu kararlılık, militanları toz pembe hayal dünyasında çok gafil avlayabilirdi.	Z _m X _{nm}