SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic

Citation for published version:
Abu Farha, I, Oprea, S, Wilson, S & Magdy, W 2022, SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic. in G Emerson, N Schluter, G Stanovsky, R Kumar, A Palmer, N Schneider, S Singh & S Ratan (eds), *Proceedings of The 16th International Workshop on Semantic Evaluation 2022.* Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 802-814, The 16th International Workshop on Semantic Evaluation 2022
, 14/07/22. https://doi.org/10.18653/v1/2022.semeval-1.111

Digital Object Identifier (DOI):
10.18653/v1/2022.semeval-1.111

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Publisher's PDF, also known as Version of record

Published In:
Proceedings of The 16th International Workshop on Semantic Evaluation 2022

SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic

Ibrahim Abu Farha[1], Silviu Vlad Oprea[1], Steven R. Wilson[1,3], Walid Magdy[1,2]
[1] School of Informatics, The University of Edinburgh, Edinburgh, UK
[2] The Alan Turing Institute, London, UK
[3] Oakland University, Rochester, MI, USA
{i.abufarha,silviu.oprea,steven.wilson,wmagdy}@ed.ac.uk

Abstract

iSarcasmEval is the first shared task to target intended sarcasm detection: the data for this task was provided and labelled by the authors of the texts themselves. Such an approach minimises the downfalls of other methods to collect sarcasm data, which rely on distant supervision or third-party annotations. The shared task contains two languages, English and Arabic, and three subtasks: sarcasm detection, sarcasm category classification, and pairwise sarcasm identification given a sarcastic sentence and its non-sarcastic rephrase. The task received submissions from 60 different teams, with the sarcasm detection task being the most popular. Most of the participating teams utilised pre-trained language models. In this paper, we provide an overview of the task, data, and participating teams.

1 Introduction

Sarcasm is a form of verbal irony that occurs when there is a discrepancy between the literal and intended meanings of an utterance. Through this discrepancy, the speaker expresses their position towards a prior proposition, often in the form of surface contempt or derogation (Wilson, 2006).

Sarcasm is present on the social web and, due to its nature, it can be disruptive of computational systems that harness this data to perform tasks such as sentiment analysis, opinion mining, author profiling, and harassment detection (Liu, 2012; Rosenthal et al., 2014; Maynard and Greenwood, 2014; Van Hee et al., 2018). In the context of SemEval, in particular, Rosenthal et al. (2014) show a significant drop in sentiment polarity classification performance when processing sarcastic tweets, compared to non-sarcastic ones. Such computational systems are widely deployed in industry, driving marketing, administration, and investment decisions (Medhat et al., 2014). In the context of Arabic, Abu Farha and Magdy (2021) show the effect of sarcasm on Arabic sentiment analysis systems, where the performance dropped significantly for the sarcastic tweets. As such, it is imperative to devise models for sarcasm detection.

Such models are usually built in a supervised learning paradigm, relying on a dataset of texts labelled as either sarcastic or non-sarcastic. Two methods have typically been used to label texts for sarcasm: distant supervision (Ptáček et al., 2014; Khodak et al., 2018; Barbieri et al., 2014), where texts are considered sarcastic if they meet predefined criteria, such as including the tag #sarcasm; or manual labelling (Filatova, 2012; Riloff et al., 2013a; Abercrombie and Hovy, 2016), where texts are presented to human annotators. However, as argued by Oprea and Magdy (2020a), both methods could produce noisy labels, in terms of both false positives, and false negatives. For instance, when human annotators label texts, they are limited by their subjective perception of sarcasm, which might differ from the intention of the authors of those texts.

In response, we suggest the current shared task, iSarcasmEval[1]. We rely on a novel method of labelling texts for sarcasm, where the sarcastic nature of texts is self-reported by the authors of those texts. Our shared task is also novel in that it includes two languages, English and Arabic, and includes three subtasks. The first subtask, covering both languages, is sarcasm detection as commonly understood: given a text, determine whether or not it is sarcastic. Next, as the sarcastic texts in our English dataset are also further labelled for the ironic speech category that they represent out of the categories specified by Leggitt and Gibbs (2000), the second subtask is: given an English text, determine which ironic speech category it represents, or whether it is non-sarcastic. Finally, for both languages, we also ask authors to provide

[1] iSarcasmEval datasets are available at: https://github.com/iabufarha/iSarcasmEval

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 802 - 814
July 14-15, 2022 ©2022 Association for Computational Linguistics

non-sarcastic rephrases of their sarcastic texts. As such, the third subtask, covering both languages, is: given a sarcastic text and its non-sarcastic rephrase, identify the sarcastic text.

We discuss related work in dataset creation, and related SemEval tasks, in Section 2. We introduce the data labelling method, and present statistics on the resulted datasets, in Section 3. We provide details on the shared task in Section 4, and on the submissions in Section 5.

2 Related Work

Most previous textual sarcasm detection datasets have been annotated using a **distant supervision** method. In this approach, texts are considered sarcastic if they meet predefined criteria, such as including specific tags (e.g. #sarcasm, #irony) (Ptáček et al., 2014; Khodak et al., 2018), or being generated by specific accounts (Barbieri et al., 2014). However, this can lead to noisy labels for several reasons. First, the tags may not mark sarcasm, but may constitute the subject or object of conversation, e.g. *"there is so much #sarcasm around!"*. Second, the assumption that certain tags always appear in conjunction with sarcasm, or that certain accounts always generate sarcasm (Barbieri et al., 2014), could lead to further false positives. Third, considering those texts that do not meet the criteria as non-sarcastic is a strong assumption that can lead to false negatives.

Due to the issues outlined above, other work has relied on **manual labelling**, where sarcasm labels are provided by human annotators (Filatova, 2012; Riloff et al., 2013a; Abercrombie and Hovy, 2016). As such, the labels represent *annotator perception*, which may actually differ from *author intention*. Annotators might lack awareness of the contextual devices that, as linguistic studies suggest (Grice, 1975; Sperber and Wilson, 1981; Utsumi, 2000), could be essential for clarifying the sarcastic intention of the authors.

Previous shared tasks in sarcasm detection (Van Hee et al., 2018; Ghanem et al., 2019; Ghosh and Muresan, 2020; Abu Farha et al., 2021) present datasets annotated via the two methods discussed above. The potential noisy labels that these methods can produce gives us reason to be concerned about the effectiveness of models that were trained on such datasets. Recently, (Shmueli et al., 2020) proposed a third method, **reactive supervision**, which aims to collect sarcastic examples based on the conversation dynamics, addressing some of these issues by using statements such as "I was being sarcastic" to automatically label texts. However, this method relies on specific cues of sarcasm which may lead to a sample that is biased toward more confusing examples that required clarification.

Further, the vast majority of sarcasm detection work (Campbell and Katz, 2012; Riloff et al., 2013b; Joshi et al., 2016; Wallace et al., 2015; Rajadesingan et al., 2015; Bamman and Smith, 2015; Amir et al., 2016; Hazarika et al., 2018; Oprea and Magdy, 2019) has focused exclusively on the English language and, due to the sociocultural aspects of sarcastic communication (Oprea and Magdy, 2020b), it is unclear if models trained on English could generalise to other languages. To our knowledge, the small amount of work on other languages such as Arabic (Karoui et al., 2017; Ghanem et al., 2019; Abbes et al., 2020; Abu-Farha and Magdy, 2020) relies on either manual labelling or distant supervision. Representative of distant supervision is the work of Karoui et al. (2017), who consider Arabic equivalents of #sarcasm, such as #سخرية, #مسخرة, and #استهزاء, to collect sarcastic tweets. Other work, (Abbes et al., 2020; Ghanem et al., 2019; Abu-Farha and Magdy, 2020; Abu Farha et al., 2021), used either manual labelling, or a mix between manual labelling and distant supervision. When working with Arabic data, these two labelling methods are even more problematic considering the large number of dialects of the language that vary both across and within countries. Relying on predefined tags in modern standard Arabic (MSA), such as those specified above, can thus lead to a plethora of false negatives. Similarly, the third-party annotators might be unfamiliar with the dialect of the texts they are annotating, resulting in erroneous manual labels.

3 Dataset

3.1 Overview

In light of the issues raised in Section 2, we propose the current shared task for sarcasm detection. We introduce a new data collection method where the sarcasm labels for texts are *provided by the authors themselves*, thus eliminating labelling proxies (in the form of predefined tags, or third-party annotators). We use this method to collect two datasets, one in English and one in Arabic.

Within each dataset, for each sarcastic text, we also ask its author to rephrase the text to convey the same intended message without using sarcasm. Finally, for the English texts, we ask trained annotators to further label each text into one of the categories of ironic speech defined by Leggitt and Gibbs (2000): sarcasm, irony, satire, understatement, overstatement, and rhetorical question. For the Arabic dataset, we also include the dialect label of the text. As such, in both datasets, each text has at least the following information attached to it: (a) a label specifying its sarcastic nature (sarcastic or non-sarcastic), provided by its author; and (b) a rephrase provided by its author that conveys the same message non-sarcastically.

3.2 Data Collection

For both English and Arabic, the sarcasm labels of texts, as well as their non-sarcastic rephrases, are provided by the authors those texts. However, the methods in which we reach authors, and how their texts are sourced, differ slightly across the two languages.

For **English texts**, we used the Prolific Academic platform[2] to recruit native English speakers who were Twitter users. We asked these participants to provide links to one sarcastic and three non-sarcastic tweets that they had posted in the past. The tweet labels were, thus, implicitly specified by the authors themselves in the process.

To collect **Arabic texts**, we were unable to find a suitable number of native Arabic speakers through Prolific Academic. Further, through our pilot study, we found that asking for tweets directly resulted in low quality data. Therefore, we used the Appen crowdsourcing platform[3] to recruit native Arabic speakers, and instead of asking for previous tweets, we asked the participants to write a short sarcastic text on the spot. Through our pilot study, we found this on-the-spot generation approach to result in high quality data. However, this methodology only provided us with sarcastic examples. As non-sarcastic examples, we used a subset of the ArSarcasm-v2 dataset (Abu Farha et al., 2021), mainly those tweets that were annotated as non-sarcastic with 100% confidence, i.e. labelled non-sarcastic by all annotators.

For each sarcastic text in **both** the English and the Arabic datasets, we also asked participants to provide an *explanation* of why the text was sarcastic, and a *rephrase* that would convey the same message non-sarcastically. For Arabic, we also collected the dialect label. We included five main dialects: Modern Standard Arabic (MSA), Gulf, Nile Basin, Levant, and North Africa.

While we asked participants to provide examples of *sarcastic* texts, we found that the provided English texts that reflected a range of different ironic speech categories, not just sarcasm. Therefore, in a second annotation stage, we paid a trained annotator to further label each English-language text with the ironic speech categories that it reflected. We adopted the categorisation presented by Leggitt and Gibbs (2000): (1) *sarcasm*: contradicts the state of affairs, is directed towards an addressee, and expresses a critical attitude; (2) *irony*: contradicts the state of affairs, may or may not be directed towards an addressee, but if it is, is not obviously critical towards that addressee; (3) *satire*: is directed towards and addressee whom it appears to support, but underneath it express disagreement, mocking, contempt, or derogation; (4) *understatement*: does not contradict the state of affairs, but undermines its weight; (5) *overstatement*: does not contradict the state of affairs, but assigns unrealistically high weight to it; (6) *rhetorical question*: a question with an implicated answer that contradicts the state of affairs. Note that these categories are not mutually exclusive. A text could belong to more than one category, e.g. it could be both sarcastic, and an understatement. Regarding Arabic, we did not go the next step to include the sarcasm categories. This is because Arabic linguists had similar disagreements regarding the differences between sarcasm and irony (Andalib and Far Shirazi, 2019). Also, it would have been a challenging task to recruit linguists who are familiar with available dialects.

3.3 Test Sets

To construct our test sets, we employed, for both languages, an approach similar to that used to collect training data in Arabic. We chose this method for collecting English test data due to restrictions that were imposed on us by the Prolific Academic crowdsourcing platform on the collection of tweets that belonged directly to survey participants.

3.4 Quality Control

For English, we made sure all tweets were posted at least 48 hours before the survey submission, and came from the same account. Further, a trained an-

[2] https://prolific.co

[3] https://appen.com

Language	**Sarcastic text**	**Unsarcastic rephrased**
EN	Gotta love people who follow you and unfollow because you don’t follow them within in an hour or 2. Sorry I don’t stay on Twitter 24/7.	I dislike people who follow me, only to unfollow me when I don’t follow back right away. I’m not on Twitter that much to follow right away.
AR	محمد صلاح ينقذ سمكة من الغرق، الله على اخلاقك يا فخر العرب (Mo Salah saves a fish from drowning. Amazing manners, you Arabs’ pride)	محمد صلاح يحمل سمكة (Mohammad Salah holds a fish)

Table 1: Examples of sarcastic tweets (tweet text) from our English and Arabic dataset along with the rephrase that authors gave that convey the same meaning non-sarcastically (rephrased).

notator consulted all survey responses provided and filtered out spurious sarcastic examples that were either unlikely to reflect sarcasm, or had uninformative explanations as to why they were sarcastic.

For Arabic, the data collection was run multiple times during a period of 8 months. In this stage, we managed to collect around 2,000 sarcastic sentences. After manual inspection, we noticed that a large portion of the texts were not truly sarcastic, or that the non-sarcastic phrasing was not informative. Thus, we hired native speakers for each of the dialects to check texts for sarcasm, and to provide or improve the non-sarcastic phrasing, if needed.

4 iSarcasmEval Details

4.1 Task Description

We formulate three subtasks:

- **Subtask A - Sarcasm Detection**: Given a text, determine whether it is sarcastic or non-sarcastic;
- **Subtask B - Sarcasm Category Classification**: Given a text, determine which ironic speech categories it belongs to, if any;
- **Subtask C - Pairwise Sarcasm Identification**: Given a sarcastic text and its non-sarcastic rephrase, i.e. two texts that convey the same meaning, determine which is the sarcastic one.

Subtasks A and C are suggested for both languages. Subtask B is only suggested for English, as we only have ironic speech category labels for English texts.

4.2 iSarcasmEval dataset

The datasets for both languages are provided as a list of texts. Each text is accompanied by a sarcasm label, indicating whether or not it is sarcastic. For sarcastic texts, there is a rephrase that conveys the same message non-sarcastically. For English sarcastic texts, there is a label specifying the category of ironic speech that it reflects. For Arabic texts, there is a label specifying the dialect. Table 1 shows a sample from our datasets, one in English, and one in Arabic. For English, the training set contains 867 and 2,601 sarcastic and non-sarcastic texts, respectively. Recall that each sarcastic text has an associated non-sarcastic rephrase. These 867 rephrases can be used as additional non-sarcastic examples. The division of sarcastic texts into ironic speech categories in the training set is shown in Table 4. There is a separate test set for each subtask. As such, the test set for subtask A contains 200 sarcastic texts, and a total of 1,200 non-sarcastic texts. The same texts, in the same arrangement, constitute the test set for subtask B. The test set for subtask C contains 200 sarcastic texts, along with their 200 non-sarcastic rephrases. These are presented as pairs, the task being to distinguish the sarcastic text from its rephrase. This information is summarised in Table 3 (training set, and test sets for subtasks A and C); and in Table 4 (training set, and test set for subtask B).

split	**total**	**sarcastic**	**non-sarcastic**
train	3,103	745	2,358
test (subtask A)	1,400	200	1,200
test (subtask C)	400	200	200

Table 2: Statistics for the Arabic training set, and test sets for subtasks A and C, as discussed in Section 4.2.

split	**total**	**sarcastic**	**non-sarcastic**
train	4,335	867	3,468
test (subtask A)	1,400	200	1,200
test (subtask C)	400	200	200

Table 3: Statistics for the English training set, and test sets for subtasks A and C, as discussed in Section 4.2.

For Arabic, the training set contains 3,103 texts, 745 of which are sarcastic. Similar to English, the sarcastic text have their non-sarcastic phrasing too. The test sets are the same size as the English test sets for both subtasks A and C. Table 2 provides a summary of the Arabic dataset splits. Table 5 provides the distribution of the whole dataset over the available dialects. It is noticeable that the majority of the sarcastic examples are in the Egyptian dialect (Nile Basin). In the future, we hope to have a higher coverage of the other dialects.

split	sarcasm	irony	satire	underst.	overst.	rhet. quest.
train	713	155	25	10	40	101
test (subtask B)	180	20	49	1	10	11

Table 4: Statistics for the English training set, and test set for subtask B, as discussed in Section 4.2.

dialect	total	sarcastic	non-sarcastic
MSA	2,035	82	1,953
Nile Basin	2,072	827	1,245
Levant	322	76	246
Gulf	278	36	242
North Africa	195	124	71

Table 5: Distribution of the Arabic dataset over the dialects.

4.3 Evaluation Metrics

The main evaluation metric for subtask A is the F1-score of the sarcastic class, referred to as $F_1^{\text{sarcastic}}$. It is computed as follows:

$$F_1^{sarcastic} = 2 \cdot \frac{P^{sarcastic} \cdot R^{sarcastic}}{P^{sarcastic} + R^{sarcastic}}, \tag{1}$$

Where $P^{sarcastic}$, $R^{sarcastic}$ are the precision and recall with respect to the sarcastic class, respectively.

For subtask B, the main evaluation metric is the macro-F1 score over all the categories of ironic speech:

$$F_1 = \frac{1}{n}\sum_{c=1}^{n}(F_1^c) \tag{2}$$

Where F_1^c represents the F_1 score for the cth category and n is the number of categories.

For subtask C, the main evaluation metric is accuracy. This is appropriate since we have an equal number of sarcastic and non-sarcastic examples.

$$Accuracy = \frac{C}{N} \tag{3}$$

Where C is the total number of correct predictions and N is the total number of pairs of text.

5 Participating Teams

5.1 Overview

The shared task saw the participation of 60 unique teams. The most popular task was subtask A (sarcasm detection) with 43 participants for English and 32 for Arabic. Subtask B received 22 submissions and subtask C received 16 submissions for English and 13 for Arabic. The following sections provide an overview of the top teams' approaches.

5.2 Subtask A (Sarcasm Detection) - English

Table 6 shows the results for English. We created two baseline models for subtask A. The first one uses the BERT language model (Devlin et al., 2019) to produce contextual representations of the input text, and considers the embedding corresponding to the [CLS] token as the aggregated representation of the input. Finally, this is provided to a classification head whose output we interpret as the probability that the input is sarcastic. We use the implementation provided as part of the transformers library Wolf et al. (2020), and initialise the encoder with the `bert-base-uncased` checkpoint published on the Huggingface model hub [4]. We fine-tune it for a maximum of 100 epochs, but use early stopping regularisation with a patience of 3. We use a learning rate of $5e-5$, and clip the norm of the gradients to 1. This results in a baseline $F_1^{\text{sarcastic}}$ of 0.348, listed as baseline-bert in Table 6. The second baseline uses a support vector machine (SVM), with a polynomial kernel of degree 3, to classify tf-idf representations of input texts. This results in a baseline $F_1^{\text{sarcastic}}$ of 0.275, listed as baseline-svm in Table 6. For both baselines, we consider the rephrases as additional non-sarcastic examples. In a preprocessing step, we remove all hashtags and urls, and replace user handles with the token `@user`.

As shown in Table 6, the team ranking first, `stce` (Yuan et al., 2022), achieved an $F_1^{\text{sarcastic}}$ of 0.605. They use an ensemble learning approach with a combination of hard and soft voting between three models, all based on the transformer architecture: RoBERTa (Liu et al., 2019), initialised with the `roberta-large` checkpoint; DeBERTa (He et al., 2021), initialised with the `deberta-v3-large` checkpoint; and XLM-RoBERTa (Conneau et al., 2020), initialised with the `xlm-roberta-large` checkpoint. XLM-RoBERTa is employed to make use of the Arabic training data for informing the classification of English texts. They experiment with several strategies to achieve their results. First, in addition to the task dataset, they also consider public datasets, including iSarcasm (Oprea and Magdy, 2020a), the dataset published by Van Hee et al. (2018), and a sample of texts from the multimodal sarcasm dataset [5]. Second, they extract statistical

[4] https://huggingface.co

[5] https://github.com/headacheboy/data-of-multimodal-sarcasm-detection

and text features that they concatenate to the text itself before providing it to the models above, such as emoji and part-of-speech information. They also use multi-sample dropout, contrastive loss functions, and adversarial training.

The team ranking second, `X-PuDu` (Han et al., 2022), achieved an $F_1^{\text{sarcastic}}$ of 0.569. They ensemble two transformer-based models: ERNIE-M (Ouyang et al., 2021), and DeBERTa, mentioned above. After providing the input text to the models, they consider the embedding corresponding to the [CLS] token as the representation of the input, which they provide to a classification head. The final ensemble considers not just the individual architectures above, but also the same architecture under different hyperparameter configurations. Using ERNIE-M, they train on both English and Arabic at the same time.

The team ranking third, `TUG-CIC` (Aroyehun et al., 2022), achieved an $F_1^{\text{sarcastic}}$ of 0.530. They use the BERT model mentioned above, but initialised with different `BERTweet` checkpoints, which they fine-tune on the SPIRS sarcasm dataset (Shmueli et al., 2020), before fine-tuning it on the data provided here. They also apply label smoothing.

5.3 Subtask A (Sarcasm Detection) - Arabic

As mentioned previously, the main metric for subtask A is the F-score of the sarcastic class. Table 7 shows the results for Arabic. The participating teams made extensive use of Arabic pre-trained language models such as MARBERT (Abdul-Mageed et al., 2021). We created two baselines for this task, the first is a Bert-based model and the other is an SVM model. We fine-tuned MARBERT for 6 epochs with a learning rate of 5e-5. For the SVM model, we used uni-gram features. Both models were trained without the non-sarcastic phrasing.

As shown in the Table 7, the top team `CS-UM6P` (El Mahdaouy et al., 2022a) achieved an $F_1^{\text{sarcastic}}$ of 0.563. This team utilised a transformer encoder (MARBERT), attention layer, and a classifier. They applied the attention to the contextualised embeddings. The classifier, which is composed of one hidden layer, is fed the concatenation of the pooled output of the encoder and the attention's output. The official submission was an ensemble of two variants of this model that are trained with and without the non-sarcastic rephrasing. `AlexU-AL` (Lotfy et al., 2022) achieved the second place with an $F_1^{\text{sarcastic}}$ of 0.508. Their model is similar to our baseline where the fine-tuned MARBERT for text classification. The results are quite close to our baseline with a small difference that can be attributed to the choice of hyper-parameters. The third team, `remarchka` (Abdel-Salam, 2022), also used MARBERT in a similar way to the baseline and `AlexU-AL` team. Their results are quite close to the other two models with $F_1^{\text{sarcastic}}$ of 0.477. The other teams followed a similar approach where they utilise one of the many flavours of Arabic-specific models or the multilingual ones. A few of the participants relied on hand-engineered features along with conventional classifiers such as SVM and Decision Trees.

r	Team Name	Affiliation	$F_1^{\text{sarcastic}}$
1	stce	PALI Inc., China	0.605
2	X-PuDu	Baidu & Shanghai Pudong Development Bank, China	0.569
3	TUG-CIC	TU Graz, Austria	0.530
4	Plumeria	Indian Institute of Technology Kanpur, India	0.477
5	John Thomson	University of Alberta, Canada	0.456
6	Naive	Dalian University of Technology, China	0.452
7	MarSan_AI	Part AI Research Center, Iran	0.434
8	LISACTeam	Sidi Mohamed Ben Abdellah University, Morocco	0.429
9	LT3	Ghent University, Belgium	0.424
10	niksss	-	0.402
11	Amobee	-	0.401
12	YNU-HPCC	Yunnan University, China	0.392
13	Dartmouth	Dartmouth College, USA	0.386
14	underfined	Ping An Life Insurance Company of China, China	0.383
15	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.371
16	UTNLP	University of Tehran, Iran	0.369
17	Jumana-Safa	-	0.356
18	cnxup	University of Chinese Academy of Sciences, China	0.351
-	baseline-bert	-	0.348
19	IISERB Brains	Indian Institute of Science Education and Research, Bhopal, India	0.345
20	rematchka	Cairo University, Egypt	0.341
21	R2D2	Vellore Institute of Technology, India	0.328
22	AMI_UofA	University of Alberta, Canada	0.312
23	Amrita-CEN	Amrita Vishwa Vidyapeetham, India	0.308
24	DUCS	University of Delhi, India	0.307
25	Happy New Year	-	0.276
-	baseline-svm	-	0.275
26	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.270
27	TechSSN	Sri Sivasubramaniya Nadar College of Engineering, India	0.264
28	NULL	Auburn University, USA	0.260
29	Cyborgs	-	0.248
30	I2C	Universidad de Huelva, Spain	0.245
31	MaChAmp	IT University of Copenhagen, Denmark	0.241
32	ISD	Stanford University, USA	0.240
33	SPDB	-	0.215
34	xuyt3	-	0.215
35	MACHON	Jerusalem College of Technology, Israel	0.215
36	FII_UAIC	University of Iasi, Romania	0.207
37	connotation_clashers	University of Tübingen, Germany	0.202
38	GetSmartMSEC	Meenakshi Sundararajan Engineering College, Chennai, India	0.201
39	UoR-NCL	University of Reading, UK	0.195
40	JCT	Jerusalem College of Technology, Israel	0.184
41	UMUTeam	Universidad de Murcia, Spain	0.180
42	MACHON	Jerusalem College of Technology, Israel	0.168
43	NARD@KGP	IIT Kharagpur, India	0.155

Table 6: Subtask A (English) results in descending order according to the main metric ($F_1^{\text{sarcastic}}$). The table shows the teams' names, rank, affiliation, and score.

5.4 Subtask B (Sarcasm Category Classification)

Table 8 shows the results. We created two baseline models for subtask B. The first baseline, listed as baseline-majority in the table, always predicts that the input reflects the ironic speech category of sarcasm, and no other category. This was chosen as it is dominant in the training set, as seen in Table 4. As a second baseline, we use the BERT language model to produce contextual representa-

r	Team Name	Affiliation	$F_1^{\text{sarcastic}}$
1	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.563
2	AlexU-AL	Alexandria University, Alexandria, Egypt	0.508
-	baseline-bert	-	0.480
3	rematchka	Cairo University, Egypt	0.477
4	HIGH-TECH Team	High Technology School, Morcco	0.468
5	Naive	Dalian University of Technology, China	0.461
6	akaBERT	Helwan University, Egypt	0.444
7	SarcasmDet	Jordan University of Science and Technology	0.431
8	Alexa	Open-Insights, Tarjamah	0.420
9	X-PuDu	Baidu & Shanghai Pudong Development Bank, China	0.419
10	Plumeria	Indian Institute of Technology Kanpur, India	0.407
11	niksss	-	0.400
12	MaChAmp	IT University of Copenhagen, Denmark	0.396
13	underfined	Ping An Life Insurance Company of China, China	0.378
14	BFCAI	Benha University	0.375
15	AM	Alexandria University,Egypt	0.369
16	cnxup	University of Chinese Academy of Sciences, China	0.367
17	stce	PALI Inc., China	0.367
18	NULL	Auburn University, USA	0.358
19	Dartmouth	Dartmouth College, USA	0.350
20	Amrita-CEN	Amrita Vishwa Vidyapeetham, India	0.349
21	YNU-HPCC	Yunnan University, China	0.323
22	UMUTeam	Universidad de Murcia, Spain	0.318
23	connotation_clashers	University of Tübingen, Germany	0.301
24	LEV	Jerusalem College of Technology, Israel	0.295
25	NARD@KGP	IIT Kharagpur, India	0.281
26	JCT	Jerusalem College of Technology, Israel	0.257
27	MACHON	Jerusalem College of Technology, Israel	0.256
28	iaf7	-	0.229
29	TechSSN	Sri Sivasubramaniya Nadar College of Engineering, India	0.229
30	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.192
31	MarSan_AI	Part AI Research Center, Iran	0.188
-	baseline-svm	-	0.139
32	UoR-NCL	University of Reading, UK	0.115

Table 7: Subtask A (Arabic) results in descending order according to the main metric ($F_1^{\text{sarcastic}}$). The table shows the teams' names, rank, affiliation, and score.

r	Team Name	Affiliation	macro F-score
1	PALI-NLP	Ping An, China	0.1630
2	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.0875
3	MaChAmp	IT University of Copenhagen, Denmark	0.0851
4	Naive	Dalian University of Technology, China	0.0809
5	X-PuDu	Baidu & Shanghai Pudong Development Bank, China	0.0799
6	Plumeria	Indian Institute of Technology Kanpur, India	0.0778
7	R2D2	Vellore Institute of Technology, India	0.0760
8	IISERB Brains	Indian Institute of Science Education and Research, India	0.0751
9	MarSan_AI	Part AI Research Center, Iran	0.0743
10	I2C	Universidad de Huelva, Spain	0.0699
11	YNU-HPCC	Yunnan University, China	0.0646
12	John Thomson	University of Alberta, Canada	0.0601
13	AMI_UofA	University of Alberta, Canada	0.0601
14	Dartmouth	Dartmouth College, USA	0.0590
15	Amrita-CEN	Amrita Vishwa Vidyapeetham, India	0.0567
16	rematchka	Cairo University, Egypt	0.0560
17	TechSSN	Sri Sivasubramaniya Nadar College of Engineering, India	0.0465
18	NARD@KGP	IIT Kharagpur, India	0.0446
-	baseline-bert	-	0.0431
19	GetSmartMSEC	Meenakshi Sundararajan Engineering College, Chennai, India	0.0387
20	niksss	-	0.0380
-	baseline-majority	-	0.0380
21	Suhaib-Aburaidah	-	0.0346
22	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.0313

Table 8: Subtask B results in descending order according to the main metric (macro F-score). The table shows the teams' names, rank, affiliation, and score.

tions of the input text, and consider the [CLS] embedding. We provide this to a classification head with a 6-dimensional output, one corresponding to each category of ironic speech. We apply the sigmoid function to each unit in the classification head, interpreting the output as the probability that the input text reflects the ironic speech category corresponding to that unit. We fine-tune the model in a similar setting as we did for subtask A. This results in a baseline macro F-score of 0.0431, listed as baseline-bert in Table 8[6].

As shown in Table 8, the team ranking first, `PALI-NLP` (Du et al., 2022), achieved a macro F-score of 0.1630. They use an ensemble learning approach, where the weight assigned to a model corresponds to its performance on a validation set. The models they consider are BERT, initialised with the `BERT-base` checkpoint; RoBERTa, initialised with the `RoBERTa-base` checkpoint; and BERTweet, initialised with the `BERTweet-base` checkpoint. Models have a classification head attached that inputs the embedding corresponding to the [CLS] token. They also use adversarial training and multi-sample dropout to improve generalisation.

The team ranking second, `CS-UM6P` (El Mahdaouy et al., 2022b), achieved a macro F-score of 0.0875. They use a model similar to GAN-BERT (Croce et al., 2020). It uses a generator that, conditioned on an ironic speech category, produces fake embeddings from a random noise that would resemble representations of examples from that ironic speech category. A discriminator is trained to recognise real examples from fake ones, while the generator is trained to cause the discriminator to classify fake examples as real. The discriminator is also trained to classify the real examples as either sarcastic, or non-sarcastic.

The team ranking third, `MaChAmp`, achieved a macro F-score of 0.0851. They first we pre-train a RemBERT (Chung et al., 2020) multi-task model across all the tasks. Then, they re-train a model for each task individually. They use the hyperparameters of `MaChAmp v0.3`(van der Goot et al., 2021), which were finetuned on the xTREME benchmark (Hu et al., 2020).

5.5 Subtask C (Pairwise Sarcasm Identification) - English

Table 9 shows the results for English. We used baselines similar to those from subtask A, but modified the input. Specifically, given a sarcastic text and its rephrase, we produced two training examples. The first was the concatenation of the sarcastic text and the rephrase, in this order, separated by a [SEP] token. This example had label 0, indicating the position of the sarcastic text. The second example was the concatenation of the rephrase and the sarcastic text, in this order, and had label 1. The first baseline, shown as baseline-bert in Table 9, achieves an accuracy of 0.765, while the second

[6] The complete results are available in Table 11 in Appendix A. Those include the scores over each sarcasm category.

r	Team name	Affiliation	Accuracy
1	X-PuDu	Baidu, China	0.870
2	Naive	Dalian University of Technology, China	0.855
3	YNU-HPCC	Yunnan University, China	0.805
4	Plumeria	Indian Institute of Technology Kanpur, India	0.790
5	LISACTeam	Sidi Mohamed Ben Abdellah University, Morocco	0.775
6	UTNLP	University of Tehran, Iran	0.770
7	MarSan_AI	Part AI Research Center, Iran	0.765
-	baseline-bert	-	0.765
8	R2D2	Vellore Institute of Technology, India	0.750
9	NARD@KGP	IIT Kharagpur, India	0.735
10	rematchka	Cairo University, Egypt	0.720
11	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.695
12	Dartmouth	Dartmouth College, USA	0.660
13	IISERB Brains	Indian Institute of Science Education and Research, Bhopal, India	0.625
14	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.495
-	baseline-svm	-	0.495
15	GetSmartMSEC	Meenakshi Sundararajan Engineering College, Chennai, India	0.340
16	MaChAmp	IT University of Copenhagen, Denmark	0.250

Table 9: Subtask C (English) results in descending order according to the main metric (accuracy). The table shows the teams' names, rank, affiliation, and score.

r	Team Name	Affiliation	Accuracy
1	Naive	Dalian University of Technology, China	0.930
2	HIGH-TECH Team	High Technology School, Morocco	0.885
3	MarSan_AI	Part AI Research Center, Iran	0.875
4	Plumeria	Indian Institute of Technology Kanpur, India	0.870
5	X-PuDu	Baidu & Shanghai Pudong Development Bank, China	0.840
6	rematchka	Cairo University, Egypt	0.800
-	baseline-bert	-	0.800
7	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.780
8	YNU-HPCC	Yunnan University, China	0.755
9	AlexU-AL	Alexandria University, Alexandria, Egypt	0.745
10	Dartmouth	Dartmouth College, USA	0.680
11	NARD@KGP	IIT Kharagpur, India	0.665
-	baseline-svm	-	0.585
12	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.465
13	MaChAmp	IT University of Copenhagen, Denmark	0.200

Table 10: Subtask C (Arabic) results in descending order according to the main metric (accuracy). The table shows the teams' names, rank, affiliation, and score.

one, baseline-svm, achieves 0.495.

As shown in Table 9, the team ranking first, X-PuDu (Han et al., 2022), achieved an accuracy of 0.870. The same team ranked second for task A, and the approach here is rather similar, except for representing the input as we do above. The team ranking second, Naive, achieved an accuracy of 0.855. They used a RoBERTa model, initialised with the RoBERTa-large checkpoint, with a classification head appended. The team ranking third, YNU-HPCC (Zheng et al., 2022), achieved an accuracy of 0.805. They also used a RoBERTa model. They did not use any external datasets during training. We suspect the difference in performance between the second and third teams to be, at least in part, the result of data preprocessing and hyperparameter optimisation.

5.6 Subtask C (Pairwise Sarcasm Identification) - Arabic

Table 10 shows the results of this task. To prepare the baselines, we utilised the models from subtask A. Since the task is to decide which text is sarcastic out of the given pair, we ran the models from subtask A on each sentence and chose the one that had a higher probability of being sarcastic. The top team, Naive (Zefeng et al., 2022), achieved an accuracy of 0.930. They utilised the model created for subtask A, where they would compare the probabilities for each sentence and choose the one with a higher probability. Their model in subtask A relied on the voting of a 5 folds cross-validation of a Bert model. High-Tech team (Hamza et al., 2022) achieved the second place with an accuracy of 0.885. They fine-tuned AraBERT (Antoun et al., 2020) on the concatenation of the sarcastic sentence and its non-sarcastic phrasing. The third team, MarSan_AI (Najafi and Tavan, 2022), achieved an accuracy of 0.875. Their model consisted of a T5 encoder (Raffel et al., 2020) followed by a transformer and Bi-LSTM, the output of the Bi-LSTM is fed to an attention layer followed by a fully connected layer. The final prediction is the softmax of the output from the fully connected layer. The other teams followed the same trend where they utilised the models from subtask A for this task. Most of these models are transformer-based models such as MARBERT and AraBERT.

In general, it is noticeable that the results on Arabic are slightly higher than the ones on English. This can be due to the slight difference in the nature of the data. As mentioned in Section 3.2, the English data are original tweets that the authors wrote before our data collection process. The Arabic data was collected on the fly, and therefore more likely to contain clear signs of sarcasm as the authors were specifically asked to provide new sarcastic and non-sarcastic phrasings.

6 Conclusion

This paper provides an overview of SemEval-2022 task 6, iSarcasmEval, which targets intended sarcasm detection. We provide an overview of the current state of research on sarcasm detection focusing on data collection methods. We introduce two new datasets for sarcasm detection in English and Arabic. The data was collected by asking people to provide and label their own words as sarcastic or not, hence intended sarcasm. iSarcasmEval contains three subtasks: sarcasm detection, sarcasm category classification, and sarcasm identification given a pair of sentences. The task was quite popular with the participation of around 62 teams. In this paper, we provide a high-level overview of the approaches of top teams in each of the subtasks. Transformer models were dominant in all subtasks.

Detecting sarcasm in texts remains challenging; detecting the ironic speech category even more so. We hope our shared task will draw the attention of the community towards these important tasks. We suggest two main directions that future work could consider.

First, in this shared task, sarcasm detection was performed by solely mining lexical and pragmatic cues from the texts being classified. However, the sarcastic intention of the authors might be unclear without reference to their previous utterances, and their sociocultural background (Oprea and Magdy, 2020b). We suggest future datasets are needed to provide access to such information, and future models that account for it effectively.

Second, the low performance achieved by the models on subtask B requires further investigation. First, alternative categorisations could be considered. Second, the ironic speech category labels should either be provided by the authors themselves, to avoid any bias introduced by trained annotators, or more emphasis should be placed on annotator training and annotation guideline clarity, to mitigate labelling noise that might indeed account, at least in part, for the low performance presented here. Finally, more effort is needed to develop more effective models, likely making use of information outside of the texts being classified, including prior assumptions about the nature of ironic speech, sociocultural information about the authors, if available, as well as commonsense facts.

7 Acknowledgements

This work was partially supported by the Defence and Security Programme at the Alan Turing Institute, funded by the UK Government; the EPSRC Centre for Doctoral Training in Data Science, funded by the UK Engineering and Physical Sciences Research Council (grant EP/L016427/1); the University of Edinburgh; and The Financial Times.

References

Ines Abbes, Wajdi Zaghouani, Omaima El-Hardlo, and Faten Ashour. 2020. DAICT: A dialectal Arabic irony corpus extracted from Twitter. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6265–6271, Marseille, France. European Language Resources Association.

Reem Abdel-Salam. 2022. reamtchka at semeval-2022 task 6: Investigating the effect of different loss functions for sarcasm detection for unbalanced datasets. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Gavin Abercrombie and Dirk Hovy. 2016. Putting Sarcasm Detection into Context: The Effects of Class Imbalance and Manual Labelling on Supervised Machine Classification of Twitter Conversations. In *Proceedings of the ACL 2016 Student Research Workshop*, pages 107–113. ACL.

Ibrahim Abu-Farha and Walid Magdy. 2020. From arabic sentiment analysis to sarcasm detection: The arsarcasm dataset. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT)*. European Language Resources Association (ELRA).

Ibrahim Abu Farha and Walid Magdy. 2021. A comparative study of effective approaches for arabic sentiment analysis. *Information Processing & Management*, 58(2):102438.

Ibrahim Abu Farha, Wajdi Zaghouani, and Walid Magdy. 2021. Overview of the wanlp 2021 shared task on sarcasm and sentiment detection in arabic. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 296–305, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mario J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. In *CoNLL*, pages 167–177. ACL.

Ali Andalib and Seyyed Heydar Far Shirazi. 2019. Controversy over the concept of irony (=al-mophareqeh) from sarcasm to contradiction; a linguistic and semantic approach. *Researches in Arabic language*, 11(20):121–134.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Segun Aroyehun, Jason Angel, and Alexander Gelbukh. 2022. Tug-cic at semeval-2021 task 6: Two-stage fine-tuning for intended sarcasm detection. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

David Bamman and Noah A. Smith. 2015. Contextualized sarcasm detection on twitter. In *ICWSM*, pages 574–577. AAAI Press.

Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. 2014. Italian irony detection in twitter: a first approach. In *CLiC-it*, page 28. AILC.

John D Campbell and Albert N Katz. 2012. Are there necessary conditions for inducing a sense of sarcastic irony? *Discourse Processes*, 49(6):459–480.

Hyung Won Chung, Thibault Févry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2020. Rethinking embedding coupling in pre-trained language models. *CoRR*, abs/2010.12821.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Danilo Croce, Giuseppe Castellucci, and Roberto Basili. 2020. GAN-BERT: Generative adversarial learning for robust text classification with a bunch of labeled examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2114–2119, Online. Association for Computational Linguistics.

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.

Xiyang Du, Dou Hu, JIN MEI ZHI, Lianxin Jiang, and Xiaofeng Shi. 2022. Pali-nlp at semeval-2022 task 6: isarcasmeval- fine-tuning the pre-trained model for detecting intended sarcasm. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Abdelkader El Mahdaouy, Abdellah El Mekki, Kabil Essefar, Abderrahman Skiredj, and Ismail Berrada. 2022a. Cs-um6p at semeval-2022 task 6: Transformer-based models for intended sarcasm detection in english and arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Abdelkader El Mahdaouy, Abdellah EL MEKKI, Kabil Essefar, Abderrahman Skiredj, and Ismail Berrada. 2022b. Cs-um6p at semeval-2022 task 6: Transformer-based models for intended sarcasm detection in english and arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *LREC*. ELRA.

Bilal Ghanem, Jihen Karoui, Farah Benamara, Véronique Moriceau, and Paolo Rosso. 2019. Idat at fire2019: Overview of the track on irony detection in arabic tweets. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, pages 10–13.

Debanjan Ghosh and Smaranda Muresan. 2020. Figlang2020 - sarcasm detection shared task.

H. P. Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press.

Alami Hamza, Abdessamad Benlahbib, and Alami Ahmed. 2022. High tech team at semeval-2022 task 6: Intended sarcasm detection for arabic texts. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Yaqian Han, Yekun Chai, Shuohuan Wang, Yu Sun, Hongyi Huang, Guanghao Chen, Yitong Xu, and Yang Yang. 2022. X-pudu at semeval-2022 task 6: Multilingual learning for english and arabic sarcasm detection. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. Cascade: Contextual sarcasm detection in online discussion forums. In *COLING*, pages 1837–1848. ACL.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

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 generalization. *CoRR*, abs/2003.11080.

Aditya Joshi, Vaibhav Tripathi, Kevin Patel, Pushpak Bhattacharyya, and Mark Carman. 2016. Are word embedding-based features useful for sarcasm detection? In *EMNLP*, pages 1006–1011. ACL.

Jihen Karoui, Farah Banamara Zitoune, and Veronique Moriceau. 2017. Soukhria: Towards an irony detection system for arabic in social media. *Procedia Computer Science*, 117:161–168.

Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. 2018. A large self-annotated corpus for sarcasm. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

John S Leggitt and Raymond W Gibbs. 2000. Emotional reactions to verbal irony. *Discourse processes*, 29(1):1–24.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

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.

Aya Lotfy, Marwan Torki, and Nagwa El-Makky. 2022. Alexu-al at semeval-2022 task 6: Detecting sarcasm in arabic text using deep learning techniques. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Diana Maynard and Mark Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4238–4243, Reykjavik, Iceland. European Language Resources Association (ELRA).

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4):1093–1113.

Maryam Najafi and Ehsan Tavan. 2022. Marsan at semeval-2022 task 6: isarcasm detection via t5 and sequence learners. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Silviu Oprea and Walid Magdy. 2019. Exploring author context for detecting intended vs perceived sarcasm. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2859, Florence, Italy. Association for Computational Linguistics.

Silviu Oprea and Walid Magdy. 2020a. isarcasm: A dataset of intended sarcasm. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Silviu Vlad Oprea and Walid Magdy. 2020b. The effect of sociocultural variables on sarcasm communication online. *Proc. ACM Hum.-Comput. Interact.*, 4(CSCW1).

Xuan Ouyang, Shuohuan Wang, Chao Pang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2021. ERNIE-M: Enhanced multilingual representation by aligning cross-lingual semantics with monolingual corpora. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 27–38, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *COLING*, pages 213–223. ACL.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 2015. Sarcasm detection on twitter: A behavioral modeling approach. In *WSDM*, pages 97–106. ACM.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013a. Sarcasm as contrast between a positive sentiment and negative situation. In *EMNLP*, pages 704–714. ACL.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013b. Sarcasm as contrast between a positive sentiment and negative situation. In *EMNLP*, pages 704–714. ACL.

Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. 2014. SemEval-2014 task 9: Sentiment analysis in twitter. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 73–80, Dublin, Ireland. Association for Computational Linguistics.

Boaz Shmueli, Lun-Wei Ku, and Soumya Ray. 2020. Reactive Supervision: A New Method for Collecting Sarcasm Data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2553–2559, Online. Association for Computational Linguistics.

Dan Sperber and Deirdre Wilson. 1981. Irony and the use-mention distinction. *Philosophy*, 3:143–184.

Akira Utsumi. 2000. Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony. *Journal of Pragmatics*, 32(12):1777–1806.

Rob van der Goot, Ahmet Üstün, Alan Ramponi, Ibrahim Sharaf, and Barbara Plank. 2021. Massive choice, ample tasks (MaChAmp): A toolkit for multitask learning in NLP. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 176–197, Online. Association for Computational Linguistics.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. SemEval-2018 task 3: Irony detection in English tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana. Association for Computational Linguistics.

Byron C. Wallace, Do Kook Choe, and Eugene Charniak. 2015. Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1035–1044, Beijing, China. Association for Computational Linguistics.

Deirdre Wilson. 2006. The pragmatics of verbal irony: Echo or pretence? *Lingua*, 116(10):1722–1743.

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.

Mengfei Yuan, Zhou Mengyuan, Lianxin Jiang, Yang Mo, and Xiaofeng Shi. 2022. stce at semeval-2022 task 6: Sarcasm detection in english tweets. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Li Zefeng, Yu Bingjie, Tuerxun Tunike, Li Zhaoqing, and Wang Yuhan. 2022. Naive at semeval-2022 task 6. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Guangmin Zheng, Jin Wang, and Xuejie Zhang. 2022. Ynu-hpcc at semeval-2022 task 6: Transformer-based model for intended sarcasm detection in english and arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

A Appendix A

Table 11 shows the complete results for subtask B.

r	Team Name	Affiliation(s)	macro F-score	F1-Sarcasm	F1-irony	F1-satire	F1-understatement	F1-overstatement	F1-rhetorical question
1	PALI-NLP	Ping An, China	0.1630	0.4828	0.1863	0.0667	0.0000	0.0870	0.1556
2	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.0875	0.2314	0.1622	0.0392	0.0000	0.0000	0.0923
3	MaChAmp	IT University of Copenhagen, Denmark	0.0851	0.2404	0.0567	0.1379	0.0000	0.0000	0.0755
4	Naive	Dalian University of Technology, China	0.0809	0.2370	0.1489	0.0000	0.0000	0.0000	0.0992
5	X-PuDu	Baidu & Shanghai Pudong Development Bank, China	0.0799	0.2271	0.1685	0.0000	0.0000	0.0000	0.0840
6	Plumeria	Indian Institute of Technology Kanpur, India	0.0778	0.2251	0.1266	0.0263	0.0000	0.0000	0.0889
7	R2D2	Vellore Institute of Technology, India	0.0760	0.2480	0.0323	0.1387	0.0034	0.0000	0.0339
8	IISERB Brains	Indian Institute of Science Education and Research, India	0.0751	0.2294	0.0963	0.0833	0.0000	0.0000	0.0414
9	MarSan_AI	Part AI Research Center, Iran	0.0743	0.1981	0.0653	0.0733	0.0000	0.0000	0.1091
10	I2C	Universidad de Huelva, Spain	0.0699	0.2430	0.0485	0.0000	0.0000	0.0000	0.1280
11	YNU-HPCC	Yunnan University, China	0.0646	0.2382	0.0577	0.0000	0.0000	0.0000	0.0920
12	John Thomson	University of Alberta, Canada	0.0601	0.2039	0.1569	0.0000	0.0000	0.0000	0.0000
13	AMI_UofA	University of Alberta, Canada	0.0601	0.2039	0.1569	0.0000	0.0000	0.0000	0.0000
14	Dartmouth	Dartmouth College, USA	0.0590	0.2293	0.0202	0.0824	0.0000	0.0077	0.0143
15	Amrita-CEN	Amrita Vishwa Vidyapeetham, India	0.0567	0.2180	0.0293	0.0461	0.0074	0.0245	0.0150
16	rematchka	Cairo University, Egypt	0.0560	0.2251	0.0285	0.0664	0.0000	0.0161	0.0000
17	TechSSN	Sri Sivasubramaniya Nadar College of Engineering, India	0.0465	0.2278	0.0282	0.0000	0.0000	0.0095	0.0137
18	NARD@KGP	IIT Kharagpur, India	0.0446	0.2281	0.0282	0.0000	0.0000	0.0000	0.0112
-	baseline-bert	-	0.0431	0.3130	0.1667	0.0000	0.0000	0.0000	0.0597
19	GetSmartMSEC	Meenakshi Sundararajan Engineering College, Chennai, India	0.0387	0.2321	0.0000	0.0000	0.0000	0.0000	0.0000
20	niksss	-	0.0380	0.2278	0.0000	0.0000	0.0000	0.0000	0.0000
-	baseline-majority	-	0.0380	0.2279	0.0000	0.0000	0.0000	0.0000	0.0000
21	Suhaib-Aburaidah	-	0.0346	0.2075	0.0000	0.0000	0.0000	0.0000	0.0000
22	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.0313	0.1538	0.0337	0.0000	0.0000	0.0000	0.0000

Table 11: Subtask B results in descending order according to the main metric (macro F-score). The table shows the teams' names, rank, affiliation, and score for each class.