

Multilingual Irony Detection with Dependency Syntax and Neural Models

Alessandra Teresa Cignarella^{1,2}, Valerio Basile¹, Manuela Sanguinetti³, Cristina Bosco¹,
Paolo Rosso² and Farah Benamara⁴

1. Dipartimento di Informatica, Università degli Studi di Torino, Italy

2. PRHLT Research Center, Universitat Politècnica de València, Spain

3. Dipartimento di Matematica e Informatica, Università degli Studi di Cagliari, Italy

4. IRIT-CNRS, Université de Toulouse, France

{cigna,basile,bosco}@di.unito.it, manuela.sanguinetti@unica.it,

proso@dsic.upv.es, farah.benamara@irit.fr

Abstract

This paper presents an in-depth investigation of the effectiveness of dependency-based syntactic features on the irony detection task in a multilingual perspective (English, Spanish, French and Italian). It focuses on the contribution from syntactic knowledge, exploiting linguistic resources where syntax is annotated according to the *Universal Dependencies* scheme. Three distinct experimental settings are provided. In the first, a variety of syntactic dependency-based features combined with classical machine learning classifiers are explored. In the second scenario, two well-known types of word embeddings are trained on parsed data and tested against gold standard datasets. In the third setting, dependency-based syntactic features are combined into the Multilingual BERT architecture. The results suggest that fine-grained dependency-based syntactic information is informative for the detection of irony.

1 Introduction

Irony is a pragmatic phenomenon that can be defined as a sort of incongruity between the literal meaning of an utterance and its intended meaning (Grice, 1975; Sperber and Wilson, 1981; Attardo, 2000). In text processing, the challenging nature of irony is mainly due to the inherent features of the phenomenon itself, that may assume a large variety of merged forms and facets. Several semantic and syntactic devices can be used to express irony (e.g., analogy, oxymoron, paradox, euphemism and rhetorical questions), and a variety of different linguistic elements can cause the incongruity, determine the clash and play the role of irony triggers within a text.

Most computational research in irony detection is applied on social media texts and focuses primarily on content-based processing of the linguistic information using semantic or pragmatic devices, neglecting syntax. For example, Joshi et al. (2015) propose a method for sarcasm detection in tweets as the task of detecting incongruity between an observed and an expected word, while Karoui et al. (2015) rely on external common knowledge to the utterance shared by author and reader. However, in addition to these pragmatic devices, some hints about the role of syntax can be found in linguistic literature where the deviation from syntactic norms has been reported as a possible trigger of the phenomenon. For instance, Michaelis and Feng (2015) explore the usage of *split interrogatives* and of *topic-comment utterances* in English, arguing they both have a double function (they express two distinct communicative acts) and that one of those is precisely that of conveying an ironic meaning (e.g. “Who are you, Willy Wonka?”).¹ The interpretation of an ironic utterance involves not only the knowledge of the usage of the language, inventory and rules including grammar, lexis, syntax, structure and style, but also the perception of the violation of these rules. Irony tactics handle the syntagmatic patterns of discourse, the way in which items precede, follow and are related to each other. As Chakhachiro (2018) shows referring to Arabic, syntactic norms, which normally see adverbs placed after verbal phrases, can be violated to trigger irony. However, when the commonly used ironic device of topic shift occurs, lexical and syntactic signals are

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: <http://creativecommons.org/licenses/by/4.0/>.

¹Michaelis and Feng (2015) use the term “sarcasm”, but for the purposes of the present paper we can gloss over the subtle differences between the two phenomena and consider *irony* as the most general term that also includes sarcasm beneath it.

always present to mark it. The fact that the expression of opinion or the production of irony are sensitive to syntactic variations occurring in texts has been moreover confirmed by [Mahler et al. \(2017\)](#), where artificial syntactic manipulations are applied, such as those based on negations and adverbs, with the purpose of fooling sentiment analysis systems.

Some research already explored different kinds of syntactic features and their interaction in several NLP tasks, showing their effectiveness. For example, [Sidorov et al. \(2012\)](#) exploited syntactic dependency-based n-grams for general-purpose classification tasks, [Socher et al. \(2013\)](#) investigated sentiment and syntax with the development of a sentiment treebank, and [Kanayama and Iwamoto \(2020\)](#) showed a pipeline method that makes the most of syntactic structures based on Universal Dependencies, achieving high precision in sentiment detection for 17 languages. Morphology and syntax have also been proved useful in a number of other tasks, such as rumor detection ([Ghanem et al., 2019](#)), authorship attribution ([Posadas-Duran et al., 2014](#); [Sidorov et al., 2014](#)) and humor detection ([Liu et al., 2018a](#)). To the best of our knowledge, very few studies use syntactic information specifically for irony detection. Among them we cite [Cignarella and Bosco \(2019\)](#), who employed a SVC combined with shallow features based on morphology and dependency syntax outperforming strong baselines in the IroSvA 2019 irony detection shared task in Spanish variants ([Ortega-Bueno et al., 2019](#)).

This paper aims to go one step further focusing for the first time on the development of syntax-aware irony detection systems in a multilingual perspective (English, Spanish, French and Italian), providing an in-depth investigation of the impact of different sources of syntactic information when used on top of several machine learning models, including Recurrent Neural Networks and Transformers. It is important to note that our aim is not to outperform the current state of the art on monolingual irony detection, but to investigate whether irony detection based on *syntax alone* can achieve comparable results with existing systems when evaluated in *multiple languages*. We believe this is an important first step before moving to complex scenarios where both syntax and pragmatic knowledge are incorporated into deep learning models which could lead to explicitly model the syntax-pragmatic interface of irony. To this end, we propose to address the following research questions:

- **(RQ-1)** *Can morphological and syntactic knowledge be helpful in addressing the task of irony detection?* - We exploit linguistic resources syntactically annotated according to a well-known dependency-based scheme (*Universal Dependencies*² UD). The versatility of the UD format allows us to apply our approaches for the detection of irony independently of a target language. Furthermore, we experimented with UD-based word embeddings, and to the best of our knowledge, neither UD annotations nor syntax-based pre-trained embeddings have previously been used for irony detection.
- **(RQ-2)** *To what extent do ad-hoc resources in UD format (treebanks) improve irony detection performances?* - We propose three distinct experimental settings. Firstly, a variety of syntactic dependency-based features combined with classical machine learning classifiers are explored, with the aim of finding the most informative set of features for detecting the presence of irony. In the second scenario two well-known word-embedding models are tested against gold standard datasets. Finally, in the third setting, dependency-based syntactic features are combined into the Multilingual BERT architecture. Our results show that our models outperform syntax-agnostic ones, which is an important first step towards syntactically-informed irony detection.
- **(RQ-3)** *Are results obtained using syntactic features stable across different languages?* - We experiment with datasets made available from previous shared tasks on irony detection in four languages: French (DEFT 2017 ([Benamara et al., 2017](#))), English (SemEval-2018 Task 3 ([Van Hee et al., 2018](#))), Spanish (IroSvA ([Ortega-Bueno et al., 2019](#))) and Italian (IronITA ([Cignarella et al., 2018](#))). The obtained results overcome the competitive baselines and are also favorably comparable with the best results reported in shared tasks on irony for the four studied languages. On the other hand, this study takes the opportunity to enrich such existing datasets with morphological and syntactic information, as encoded in UD.

²<https://universaldependencies.org/>.

The paper is organized as follows. In the next section, related work regarding irony detection is surveyed, while Section 3 describes the data and experimental settings we followed. In Section 4, we provide a description of the experiments we performed, and, in the last section, we discuss the experiments and provide some final remarks through an error analysis. We conclude the paper with some insights about the impact of morpho-syntax information and suggest possible directions for future work.

2 Related work

The identification of irony and the description of pragmatic and linguistic devices that trigger it has always been a controversial topic (Grice, 1975; Sperber and Wilson, 1981; Utsumi, 1996), a challenge for both humans and automatic tools. This motivates the interest for irony of the NLP community, the organization of tasks regarding its processing and the subsequent creation of benchmarks.

In the last decade several shared tasks have been organized in order to advance with the techniques regarding the automatic processing of figurative language (Ghosh et al., 2015), and more recently irony and sarcasm detection have been gaining greater attention. Being irony a pragmatic device inherently related to culture and language, the task captured the attention of various researchers, dealing with different languages and mostly focusing on social media texts. The presence of irony for Italian has been investigated -among other things- in the SENTIPOLC task of the EVALITA 2014 (Basile et al., 2014) and 2016 (Barbieri et al., 2016) and at IronITA (Cignarella et al., 2018). It was addressed for French within the context of DEFT 2017 (Benamara et al., 2017), for English at Semeval 2018 (*Task 3: Irony Detection in English Tweets*) (Van Hee et al., 2018) as well as for Spanish at the IroSVA (González et al., 2019). As a side effect, several annotated corpora were released, among which also those that allowed us the composition of the experimental settings provided in this paper (see § 4).

Similar to other NLP tasks, neural approaches in irony detection have been widely used, and proved successful even in the mentioned shared tasks. For instance, the best-scoring team for Task 3 at Semeval 2018 used a densely connected LSTM based on pre-trained word embeddings, sentiment and PoS tag features (Wu et al., 2018); they built a multi-task model to predict the missing irony hashtag, whether a tweet is ironic or not and the fine-grained type of irony. A similar multi-task learning approach was adopted in Cimino et al. (2018) at IronITA 2018, which introduced a 2-layer BiLSTM that exploits additional information, such as automatically-generated sentiment polarity lexica, word-embedding lexica and PoS tags. The irony detection system proposed in González et al. (2019) at IroSVA 2019 was based on the Transformer Encoder model. At DEFT 2017 the winning team of the shared task for French (Rouvier and Bousquet, 2017) proposed an approach based on varying the size of the layers of a CNN combined with three different sentiment-based word embeddings. Finally, a multilingual scenario is also proposed in Ghanem et al. (2020), that describe both feature-based models (with Random Forest showing the best results) and a CNN architecture with multilingual embeddings, and test their approach on Arabic, English and French (for the latter, the DEFT 2017 dataset was used, as in the present work). As mentioned in Section 1, to the best of our knowledge, very few studies use syntactic information specifically for irony detection.

While in this work as well we tested our models in a multilingual setting, the novelty of the approach provided in this paper consists in using features which represent syntactic knowledge that can be extracted from text by applying morphological and syntactic analysis. Universal Dependencies played a key role in this respect: in recent years they have become a popular and widely-used annotation format (Nivre et al., 2020), providing a set of broadly attested universal grammatical relations, in which existing dependency schemes for different languages can be mapped. The proliferation of UD treebanks in a short time span highlights the need for a consistent representation of linguistic phenomena, not only for typical contrastive studies, but also when using such resources for downstream applications.

3 Data and Experimental Setting

In order to address the research questions, we follow a binary classification irony detection task, testing an automatic system on four different languages: English, Spanish, French and Italian. This will allow us to investigate whether syntactic structures, independently of the target language, are informative to

understand whether a message is ironic or not.

In the multilingual experimental setting we took advantage of four datasets that have been made available during the last few years within evaluation campaigns, and all featured by a binary annotation for irony. In Table 1, for each dataset, we report the target language, the name of the shared task in which it was released and its reference, the number of tweets for each class (*ironic* vs. *not ironic*) and the total number of instances, for both training set and test set. The aim of our task is, thus, to determine whether a given tweet is *ironic* or *not ironic*.

language	dataset	train			test		
		ironic	not	total	ironic	not	total
English	SemEval-2018 Task 3 (Van Hee et al., 2018)	1,923	1,911	3,834	311	473	784
Spanish	IroSvA 2019 (Ortega-Bueno et al., 2019)	1,600	5,600	7,200	599	1,201	1,800
French	DEFT 2017 (Benamara et al., 2017)	1,947	3,906	5,853	488	976	1,464
Italian	IronITA 2018 (Cignarella et al., 2018)	2,023	1,954	3,977	435	437	872

Table 1: Benchmark datasets used for irony detection (binary task).

In Table 2, we show a pair of ironic and non-ironic tweets for each language, with examples retrieved from the datasets used in the present study.

language	tweet	irony
English	Just great when you're mobile bill arrives by text	ironic
	i feel like whole life is about waiting waiting and waiting	not
Spanish	#Bienvenidos a #Carmenolandia. Ciudad del #Atasco perpetuo. #SemáforosA5 <i>#Welcome to #Carmenoland. City of the infinite rush hour. #TrafficLightsA5</i>	ironic
	La fotografía que ilustra los #SemáforosA5 https://t.co/qKLXgr6jJF <i>The photograph that shows the traffic lights A https://t.co/qKLXgr6jJF</i>	not
French	Hollande et Sarkozy au second tours, on s'attendez pas a ça <i>Hollande and Sarkozy in the second round election, we didn't expect it</i>	ironic
	Le football c'est pas cruel, c'est juste la loi du plus fort <i>Football is not cruel, it's just the law of the strongest</i>	not
Italian	Nuovo governo monti: dal bunga bunga al banca banca... <i>New Monti's Governmet: from bunga bunga to bank bank...</i>	ironic
	REUTERS ANALISI Crisi, di corsa verso il governo Monti <i>REUTERS ANALYSIS Crisis, running towards Monti's Government</i>	not

Table 2: Examples from the datasets.

Provided that the availability of morphological and syntactic knowledge is crucial for performing the experiments described in the rest of the paper, we needed to obtain a representation of all the datasets in UD format. With the exception of TWITTIRÒ (Cignarella et al., 2019), a subset of the IronITA dataset already available as a UD treebank (see the current release in the UD official repository³), we obtained the dependency-based annotation for the other corpora by applying the *UDPipe*⁴ pipeline (for tokenization, PoS-tagging and parsing). Figure 1 provides an example drawn from TWITTIRÒ.

Considering that all the datasets used in this work consist of Twitter data, whenever possible, we used resources where this genre, or at least user-generated content of some kind, was included as training data for parsing. More precisely, the model for English has been trained on the EWT treebank (Silveira et al., 2014), that for Spanish on both GSD-Spanish (McDonald et al., 2013) and ANCORA corpora (Taulé et al., 2008). The model for Italian – for the remaining part of IronITA⁵ – was trained on POSTWITA (Sanguinetti et al., 2018) and ISDT treebanks (Simi et al., 2014), while that for French on the GSD-French corpus (McDonald et al., 2013). We are aware that there actually exists a Twitter treebank for English, i.e. TWEEBANK V2 (Liu et al., 2018b), but it is not fully compliant with the UD format

³https://github.com/UniversalDependencies/UD_Italian-TWITTIRÒ.

⁴<http://ufal.mff.cuni.cz/udpipe>.

⁵Approximately 1,400 out of 4,849 tweets from IronITA are also part of the TWITTIRÒ corpus, already available in UD.

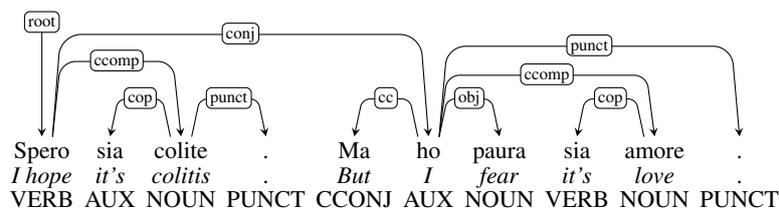


Figure 1: The dependency tree in UD format.

specifications (e.g. it violates the single root constraint posed in UD format). We thus opted for the EWT in order to preserve annotation consistency among resources.

The different amount of data used for training UDPipe, the variety of text genres among the datasets, and the fact that the UD annotation of only one dataset has undergone a manual correction – i.e. TWITTERÒ, also mentioned above – can make the quality of the UD data used in this study not entirely homogeneous. In spite of such disparity in the annotation reliability, and bearing in mind that a higher accuracy in this regard can be crucial, we considered the output provided by the parser for all the languages reasonably satisfactory for the purposes of our study.

4 Methodology

The main aim of the experiments presented in this section consists in evaluating the contribution to irony detection made by the linguistic information provided in the datasets described above. The task we propose is, therefore, a straightforward binary classification task on irony detection, that is, the task for which literature offers baselines and fair-sized annotated datasets for a variety of languages.

For addressing the task, we performed a set of experiments where several models were implemented exploiting classical machine learning algorithms, deep learning architectures and state-of-the-art language models implemented with the Python libraries *scikit-learn*⁶ and *keras*⁷. We tested different sets of pre-trained word embeddings to initialize the neural models, namely *fastText*⁸ and a dependency-based *word2vec* proposed by Levy and Goldberg (2014) (*word2vecf*). The latter was trained on the concatenation of all the treebanks available in the UD repository for each considered language.

In order to combine these methods with syntactic features inspired by Sidorov et al. (2014), we used data where not only a binary annotation for irony is applied, but also a morphological and syntactic analysis is available (see § 4).

4.1 Pre-processing and Features

We stripped all the URLs and we normalized all characters to lowercase letters, as it is often done before the application of sentiment analysis tools. We investigated the use of novel features aimed at exploiting information conveyed by syntax, studying in particular the impact of the availability of training resources in UD format.

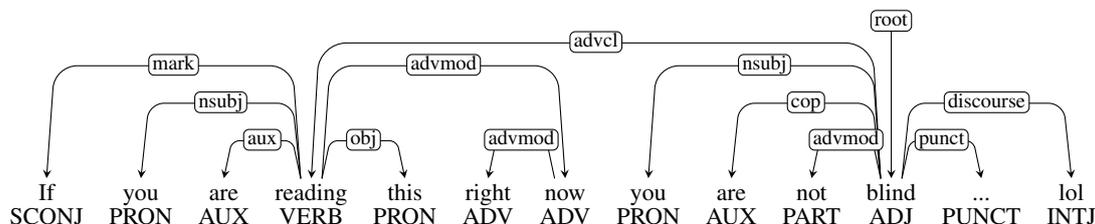


Figure 2: Dependency-based syntactic tree of an English tweet.

⁶<https://scikit-learn.org>.

⁷<https://keras.io/>.

⁸<https://fasttext.cc/>.

The description of features as well as the content of the vectors for the syntactic features we developed, referring to the tweet in Figure 2, are as follows:

- **ngrams**: we extracted unigrams, bigrams and trigrams of tokens; e.g. [*If, you, are, reading, ..., If you, you are, are reading, ..., If you are, you are reading, are reading this, ...*];
- **chagrams**: we considered the sequence of char-grams in a range from 2 to 5 characters; eg [*If, fy, yo, ou, ..., Ifyou, fyoua, youar, ouare, uarer, ...*];
- **deprelneg**: we considered the presence of negation in the text, relying on the morpho-syntactic cues present in the UD format. When a negation was present, we appended the corresponding dependency relation in the feature vector. For instance, in Figure 2, we spot a negation in [*... are not blind ...*], the dependency relation of “not” is *advmod*, therefore we append it in the feature vector;
- **deprel**: we built a bag of words of 5-grams, 6-grams and 7-grams of dependency relations as occurring in the linear order of the sentence from left to right; e.g. [*mark nsubj aux obj advmod, nsubj aux obj advmod advmod, ..., advmod advmod nsubj cop advmod root punct, advmod nsubj cop advmod root punct discourse*];
- **relationformVERB**: we create a feature vector with all the tuples of tokens that are connected with a dependency distance $DD = 1$, by starting from a verb and at the same time we blank the verb itself. For instance, in our example the first verb is “reading” and some of the tuples of tokens connected through this verb are, e.g. [*IfVERBthis, youVERBthis, areVERBthis, IfVERBnow, youVERBnow, ...*];
- **relationformNOUN**: we applied the same procedure of the feature above but considering nouns as starting points for collecting tuples;
- **relationformADJ**: in the same fashion of the two features above, we repeated the same procedure for adjectives too;
- **Sidorovbigramsform**: we created a bag of word-forms (tokens), considering the bi-grams that can be collected following the syntactic tree structure (rather than the bi-grams that can be collected reading the sentence from left to right).⁹ Such that: e.g. [*blind reading, blind you, blind are, blind not, reading if, reading you, ...*];
- **Sidorovbigramsupostag**: as the feature above, we created a bag of Part-of-Speech tags;
- **Sidorovbigramsdeprel**: as the two features above, we created a bag of words based on dependency relations (*deprels*).

4.2 Models

Having as primary goal the exploration of the features described in the previous section and as a case study testing their effectiveness in irony detection, we implemented a variety of models, including the following:

Support Vector Machine (SVM) - The choice of the kernel could be a challenging task, because the performance is dataset dependent. In this case we opted for a linear kernel, which is usually recommended for text classification.¹⁰

Logistic Regression (LR) - We used the Logistic Regression classifier with the default parameters, exception made for the maximum number of iterations which we set to 5.

Random Forest (RF) - We used the Random Forest classifier with its default parameters.

Multi-Layer Perceptron (MLP) - We used *GridSearch*¹¹ to adjust the hyper-parameters, thus obtaining: hidden layer size = 30, the activation function for the hidden layer = *logistic*, the solver for weight optimization = *lbfgs*. We later set the *early stopping* to `True`, and an initial learning rate = .01, and the size of minibatches = 5.

GRU - We used a straightforward architecture and set the following hyper-parameters: epochs = 10, learning rate = .0001. The GRU is initialized with random weights and the word embeddings are learned during the training.¹²

⁹Please refer to Sidorov et al. (2013) and Sidorov (2014) for more details on this regard.

¹⁰We also experimented with RBF kernel but it obtained worse performances due to sparse feature matrices.

¹¹<https://bit.ly/2NxU4xC>.

¹²We experimented also with other RNN architectures but they were not conclusive.

GRU+fastText - With the same hyper-parameters as above, we initialized the neural network with specific *fastText* word embedding pre-trained models (Joulin et al., 2016) for each language.

GRU+dependency-based embeddings - In the same way, and maintaining the same setting, we fed the neural network with the dependency-based *word2vec* embeddings by Levy and Goldberg (2014). These word embeddings are able to capture syntactic information during training, therefore producing embeddings that are more sensitive to functional similarity than traditional co-occurrence-based word embeddings like *fastText*. We trained a different word embedding model for each language using the concatenation of available UD treebanks for that language.

Multilingual BERT - Since a multilingual version of the BERT language model (henceforth M-BERT)¹³ is also available for 104 languages including the four we take into account in this study, we also performed a set of experiments using it. We set the hyper-parameters such that the batch size = 8, the initial learning rate = $1e - 5$. We did not set any fixed number of epochs, but rather relied on the *EarlyStopping* function, setting the value of patience = 5.

M-BERT+syntax - With the same hyper-parameters as above, we concatenated the dependency-based features from § 4.1 to those extracted from M-BERT and fed both of them into a LSTM neural network.

M-BERT+best_feats - Maintaining the same setting as the two previous models, here we concatenated a smaller set of features, and not all of them. Namely, we selected only the *best features* resulting from experiments with classical machine learning models, as we will explain in § 5.1.

M-BERT+autoencoder - Finally, we experimented by reducing the features with a technique known as *Autoencoder* (Bengio, 2009) which is becoming more and more popular in NLP (Zabalza et al., 2016).

5 Experiments and Results

We performed different sets of experiments training the models on the training sets and evaluating them against gold test sets made available in the shared tasks referred to in Table 1. We highlight three main scenarios: § 5.1 - experiments to select the best features; § 5.2 - experiments to measure the impact of dependency-based word embeddings; and finally § 5.3 - experiments conducted with the bidirectional encoder BERT in its multilingual variant infused with syntax knowledge.

5.1 Selection of Best Features

Firstly, in order to understand which of the features provided in 4.1 are relevant for the present task, we performed experiments with classical machine learning algorithms. We carried out an evaluation of four models (SVM, LR, RF and MLP) by combining them with all the possible permutations of the feature set and we evaluate them with respect to the macro-F1 score (the averaged score between the F1 of the ironic class and the F1 of the non-ironic one).¹⁴ This first step was crucial for establishing which single feature or group of combined features performs better for each of the involved languages, as summarized in Table 3.

language	macro F1	model	ngrams	chagrams	deprel	deprelneg	relation form VERB	relation form NOUN	relation form ADJ	Sidorov bigrams form	Sidorov bigrams deprel	Sidorov bigrams upostag
English	.683	RF		✓			✓		✓		✓	✓
Spanish	.539	RF							✓			
French	.641	LR		✓			✓	✓		✓		✓
Italian	.702	RF		✓	✓		✓	✓		✓	✓	

Table 3: Features exploited in the best runs with classical ML algorithms in each language scenario.

From Table 3 it emerges how in all the configurations used for achieving the best score, at least one dependency-based syntactic feature was exploited and in particular those based on Sidorov’s work, i.e. the last three columns of the table¹⁵. This provides evidence for positively answering to our first research

¹³<https://github.com/google-research/bert/blob/master/multilingual.md>.

¹⁴Supplementary material such as code, detailed features and exhaustive tables of results are accessible here: https://github.com/AleT-Cig/DependencySyntax_DeepIrony.

¹⁵Please refer to Sidorov et al. (2012) and Sidorov et al. (2013) for further details.

question, since those are the features where the real structure from root to branches of syntactic trees is encoded.

Moreover, Italian is the best scoring language (F1 = .702). This might provide some hints about our second research question, suggesting that the higher quality of the resource from where syntactic knowledge has been drawn might positively influence the performance. As previously mentioned, the one for Italian is the only dataset used for this study previously submitted to a careful manual check, after the processing done with UDPipe. On the contrary, the datasets for the other three languages are obtained automatically without further manual revision.

Interestingly, in any configuration *ngrams* and *deprelneg* are used. Regarding the latter, we noticed that it was a weak feature, and very sparse, thus not able alone to capture a complex phenomenon as irony. Concerning *ngrams*, although, the fact that they are not exploited in the best configurations might point towards the fact that to detect irony a lexical approach is not really sufficient.

Through this step we are able to consider these combinations of characteristics as “*best feature sets*” for each language and to exploit them in further experiments (see § 5.3 below in this section).

5.2 Measuring the Impact of Dependency-Based Word Embeddings

In the second set of experiments we explored, instead, the potential of two different kinds of word embeddings with regard to each language, namely *fastText* and the dependency-based word embeddings version of *word2vec*, which we trained on the four datasets taking advantage of the UD representation (*dep-based we*).

language	GRU		
	—	+fastText	+dep-based we
English	.648	.650	.552
Spanish	.494	.500	.404
French	.522	.567	.447
Italian	.649	.652	.659

Table 4: Results obtained combining a GRU architecture and word embeddings.

As depicted in Table 4, for each language, the best results in terms of macro F1 are obtained adopting either *fastText* or the *dependency-based word embeddings*. Furthermore, we highlight that *fastText* seems overall the best configuration in this set of experiments (English, Spanish and French), exception made for Italian in which the best result is obtained with the *dependency-based word embeddings* configuration. The most significant insight that we get from this second set of experiments is that neural architectures (in this case GRUs) can benefit from the addition of syntactic features and from training on UD-based representations, although they strictly depend on the quality of training data. In fact, from the results shown in Table 4, it can be seen that the performance of the dependency-based word embeddings is much lower than that of the randomly initialized word embeddings and the word embeddings pre-trained from *fastText* for most languages. A positive result with respect to this is obtained in Italian, where the *dependency-based word embeddings* configuration gives some hints about the impact that a good quality of the morpho-syntactic annotation might provide. The Italian dataset is, in fact, the only one amongst the four ones that has been manually checked and corrected and whose quality could be considered superior compared to the resources used for the other languages. Furthermore, Italian is the only language configuration in which the *dependency-based word embeddings* have been trained on UD-based text from the same textual genre (tweets). To the best of our knowledge, *dependency-based word embeddings* were not used before in the detection of irony; on the other hand, in a previous work, [MacAvaney and Zeldes \(2018\)](#) expand the findings of [Levy and Goldberg \(2014\)](#) exploring the effectiveness of various dependency-based word embeddings on domain similarity, word similarity, and two other downstream tasks in English (question-type classification and named entity recognition.). They found that embeddings trained with UD contexts excel only in some tasks but not always improve the performance.

It is our belief that fine-grained syntactic information such as the features we implemented in § 4.1, when added to an already robust architecture, could capture important structures of language and therefore boost the performance of the system. We try to make this hypothesis evident by building the third and last experimental setting, as follows.

5.3 Syntactically-informed BERT for Irony Detection

Lastly, we performed experiments with the state-of-the-art BERT language model. For each language, we ran the base cased M-BERT model as anticipated in Section 4.2. In a second phase of this setting, we implemented the base architecture by adding the dependency-based syntactic features detailed in 4.1 in three different fashions in order to have a clear-cut evidence of the actual contribution derived from dependency syntax to irony detection.

language	shared task winner (report and score)	SVC +unigrams	BERT				
			+syntax	+best_feats	+autoen.		
English	(Wu et al., 2018)	.705	.649	.655	.682 (↑ .027)	.694 (↑ .039)	.706 (↑ .051)
Spanish	(González et al., 2019)	.683	.613	.663	.668 (↑ .003)	.677 (↑ .014)	.679 (↑ .016)
French	(Rouvier and Bousquet, 2017)	.783	.617	.770	.785 (↑ .015)	.772 (↑ .002)	.679 (↓ .091)
Italian	(Cimino et al., 2018)	.731	.578	.699	.703 (↑ .004)	.687 (↓ .012)	.696 (↓ .003)

Table 5: Results obtained combining M-BERT and dependency-based syntactic features. Green values and arrows pointing up show an increment in performance, while red values and arrows pointing down indicate a performance reduction, with respect to results obtained by the bare architecture.

In Table 5 we report the results of the best participating system in each one of the shared tasks (with the reference to their working notes). Furthermore, as a baseline reference measure, we also added the results obtained with a SVC and a bag of words of unigrams, as it was a baseline proposed in all competitions. Each of the experiments with M-BERT has been performed 5 times with the hyper-parameters previously described in Section 4.2 in order to take into account the differences of random initialization, and the average macro F1 score of such number of runs is reported.

It is interesting to see that, firstly, models implemented with the addition of dependency-based syntactic features obtain results in line with the state of the art in all four language scenarios (see shared tasks results). Moreover, in all of them, the addition of syntactic knowledge (*M-BERT+syntax*) determined an improvement of scores with respect to the models where syntax is not taken into account (*M-BERT*), as highlighted with the green values. Therefore, it seems that syntax plays an important role in the detection of irony and surely deserves further investigation exploring also more complex neural architectures.

We carried out a third set of experiments, which produced the scores reported in the table as *M-BERT+best_feats*. In these experiments we paired the M-BERT architecture with the best set of features extracted in the very first setting (see Section 5.1). What we observe in this case is that the extraction of the best features, and the subsequent reduction of the dimension of the feature space, is beneficial for English and Spanish. As a matter of fact, using the reduced feature set, we have an increment of the macro F1 score from .682 to .694 (+.012) in English and from .668 to .677 (+.009) in Spanish. On the other hand, French and Italian do not seem to profit from the reduction of available features.

Lastly, we tried a different technique for reducing the sparsity of features derived from syntax knowledge, implementing an *autoencoder* (Bengio, 2009), a well-known technique to perform dimensionality reduction. The results of *M-BERT+autoencoder* for all languages are comparable to those obtained with the other settings, and especially in English they are close to the state-of-the-art results and even surpass it by few points. For French and Italian, the autoencoding technique does not seem to be beneficial at all (see red values). It obtains, indeed, the worst results among all the performances.

All the above-mentioned conditions prove themselves interesting and surely lead us in the direction of further investigation, pointing mainly towards a better understanding of features' behavior when stacked in a pre-trained language model such as BERT. Furthermore, taking into account that irony is a pragmatic

device inherently related to culture and language, our findings stress the importance of investigating features monolingually in order to provide a solid background for enhancing a multilingual system.

5.4 Error Analysis

From a lexical point of view, our qualitative analysis of results shows that even in the presence of clear lexical cues, like specific hashtags, classification errors occur in all datasets. This piece of evidence does not come unexpected. It is indeed a confirmation that our classification method –almost exclusively based on morpho-syntactic information– which we developed in order to highlight the impact of syntax alone in the task of irony detection, is not especially influenced by the presence of such cues.

On the other hand, an error analysis based on the results generated from the best runs in the four languages also provides useful evidences about the impact of syntactic features to the detection of irony. The distribution of Part-of-Speech tags and dependency relations in the whole test sets with respect to their distribution in the misclassified tweets is indeed unbalanced. Namely, a higher number of `SYM`¹⁶ and `X`¹⁷, with respect to PoS-tags, and `parataxis`¹⁸, `flat`¹⁹ and `expl`²⁰, with respect to dependency relations, characterizes tweets that are wrongly classified (either false positives or false negatives). Regardless of the variation of the frequency of these morphological categories and syntactic relations in the four different languages, their incidence can be observed in the results. For instance, the presence of tokens tagged as `SYM` in the English misclassified tweets is higher by +22% with regard to the average distribution, in Spanish is +7%, in Italian is +4%, while it does not seem to have particular relevance in the French data.

The categories and relations observed are typically found in user-generated texts and are moreover known as inherently hard to parse for the state-of-the-art UD parsers which, still nowadays, are mostly trained on standard texts and do not take into account idiosyncrasies of social media texts (emojis, hashtags, non-standard word forms). A debate regarding this one and other related issues is indeed ongoing in the UD community and a discussion can be seen, e.g., in a recent work by [Sanguinetti et al. \(2020\)](#).

6 Conclusion and Future Work

In the present paper, we have investigated different scenarios regarding irony detection exploiting dependency-based syntactic features in a multilingual perspective. Our current findings provide a meaningful support to the hypothesis that morpho-syntactic knowledge extracted from treebanks can be usefully exploited for addressing the irony detection task. In particular, they pave the way for a further investigation where the combination of a dependency-based syntactic approach and state-of-the-art neural models can be explored.

In the future, in order to validate our findings, we will propose a novel and wider experimental setting, where more languages are included, e.g. testing our approach on other languages for which both irony-annotated datasets and UD resources are available, such as Arabic ([Ghanem et al., 2020](#); [Seddah et al., 2020](#)) or German²¹ ([Rehbein et al., 2019](#)).

Acknowledgments

The work of C. Bosco and V. Basile was partially funded by Progetto di Ateneo/CSP 2016 Immigrants, Hate and Prejudice in Social Media (S1618_L2_BOSC_01). The work of M. Sanguinetti is funded by PRIN 2017 (2019-2022) project *HOPE - High quality Open data Publishing and Enrichment*. The work of P. Rosso was partially funded by the Spanish MICINN under the research project MISIMIS-FAKEHATE on Misinformation and Miscommunication in social media: FAKE news and HATE speech (PGC2018-096212-B-C31) and by the Generalitat Valenciana under the project DeepPattern (PROMETEO/2019/121). Thanks also to P. Chiril for her suggestion on implementation methods.

¹⁶See: <https://universaldependencies.org/u/pos/SYM.html>.

¹⁷See: <https://universaldependencies.org/u/pos/X.html>.

¹⁸See: <https://universaldependencies.org/u/dep/parataxis.html>.

¹⁹See: <https://universaldependencies.org/u/dep/flat.html>.

²⁰See: <https://universaldependencies.org/u/dep/expl.html>.

²¹<http://kti.tugraz.at/staff/rkern/courses/kddm2/2018/reports/team-27.pdf>.

References

- Salvatore Attardo. 2000. Irony markers and functions: Towards a goal-oriented theory of irony and its processing. *Task: International journal of language and communication*, 12(1):3–20.
- Francesco Barbieri, Valerio Basile, Danilo Croce, Malvina Nissim, Nicole Novielli, and Viviana Patti. 2016. Overview of the EVALITA 2016 SENTiment POLarity Classification task. In *Proceedings of EVALITA 2016 Evaluation of NLP and Speech Tools for Italian*. CEUR-WS.org.
- Valerio Basile, Andrea Bolioli, Malvina Nissim, Viviana Patti, and Paolo Rosso. 2014. Overview of the EVALITA 2014 SENTiment POLarity Classification task. In *Proceedings of EVALITA 2014 Evaluation of NLP and Speech Tools for Italian*. CEUR-WS.org.
- Farah Benamara, Cyril Grouin, Jihen Karoui, Véronique Moriceau, and Isabelle Robba. 2017. Analyse d’opinion et langage figuratif dans des tweets: présentation et résultats du Défi Fouille de Textes DEFT2017. In *Actes de l’atelier DEFT2017 associé à la conférence TALN*, pages 1–12. Association pour le Traitement Automatique des Langues (ATALA).
- Yoshua Bengio. 2009. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, 2(1):1–127.
- Raymond Chakhachiro. 2018. *Translating Irony between English and Arabic*. Cambridge Scholars Publishing.
- Alessandra Teresa Cignarella and Cristina Bosco. 2019. ATC at IroSvA 2019: Shallow Syntactic Dependency-based Features for Irony Detection in Spanish Variants. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019)*, pages 257–263. CEUR-WS.org.
- Alessandra Teresa Cignarella, Simona Frenda, Valerio Basile, Cristina Bosco, Viviana Patti, and Paolo Rosso. 2018. Overview of the Evalita 2018 task on irony detection in Italian tweets (IRONITA). In *Proceedings of EVALITA 2018 Evaluation of NLP and Speech Tools for Italian*. CEUR-WS.org.
- Alessandra Teresa Cignarella, Cristina Bosco, and Paolo Rosso. 2019. Presenting TWITTIRO-UD: An Italian Twitter Treebank in Universal Dependencies. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 190–197. Association for Computational Linguistics (ACL).
- Andrea Cimino, Lorenzo De Mattei, and Felice Dell’Orletta. 2018. Multi-task learning in deep neural networks at Evalita 2018. In *Proceedings of EVALITA 2018 Evaluation of NLP and Speech Tools for Italian*, pages 86–95. CEUR-WS.org.
- Bilal Ghanem, Alessandra Teresa Cignarella, Cristina Bosco, Paolo Rosso, and Francisco Manuel Rangel Pardo. 2019. UPV-28-UNITO at SemEval-2019 Task 7: Exploiting Post’s Nesting and Syntax Information for Rumor Stance Classification. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval 2019)*, pages 1125–1131. Association for Computational Linguistics (ACL).
- Bilal Ghanem, Jihen Karoui, Farah Benamara, Paolo Rosso, and Véronique Moriceau. 2020. Irony detection in a multilingual context. In *Proceedings of the European Conference on Information Retrieval*, pages 141–149. Springer.
- Aniruddha Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. 2015. SemEval-2015 task 11: Sentiment analysis of figurative language in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 470–478. Association for Computational Linguistics (ACL).
- José-Ángel González, Lluís-Felip Hurtado, and Ferran Pla. 2019. ELiRF-UPV at IroSvA: Transformer Encoders for Spanish Irony Detection. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019)*. CEUR-WS.org.
- Herbert P. Grice. 1975. Logic and conversation. In P. Cole and J. L. Morgan, editors, *Speech acts*, pages 41–58. New York: Academic Press.
- Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015. Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 757–762. Association for Computational Linguistics (ACL).
- Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Herve Jégou, and Tomas Mikolov. 2016. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

- Hiroshi Kanayama and Ran Iwamoto. 2020. How Universal are Universal Dependencies? Exploiting Syntax for Multilingual Clause-level Sentiment Detection. In *Proceedings of The 12th Language Resources and Evaluation Conference (LREC 2020)*, pages 4063–4073. European Language Resources Association (ELRA).
- Jihen Karoui, Farah Benamara Zitoune, Véronique Moriceau, Nathalie Aussenac-Gilles, and Lamia Hadrich Belguith. 2015. Towards a contextual pragmatic model to detect irony in tweets. In *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 644–650. Association for Computational Linguistics (ACL).
- Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 302–308. Association for Computational Linguistics (ACL).
- Lizhen Liu, Donghai Zhang, and Wei Song. 2018a. Exploiting syntactic structures for humor recognition. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1875–1883. Association for Computational Linguistics (ACL).
- Yijia Liu, Yi Zhu, Wanxiang Che, Bing Qin, Nathan Schneider, and Noah A. Smith. 2018b. Parsing tweets into Universal Dependencies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 965–975. Association for Computational Linguistics (ACL).
- Sean MacAvaney and Amir Zeldes. 2018. A Deeper Look into Dependency-Based Word Embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 40–45. Association for Computational Linguistics (ACL).
- Taylor Mahler, Willy Cheung, Micha Elsner, David King, Marie-Catherine de Marneffe, Cory Shain, Symon Stevens-Guille, and Michael White. 2017. Breaking NLP: Using Morphosyntax, Semantics, Pragmatics and World Knowledge to Fool Sentiment Analysis Systems. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 33–39. Association for Computational Linguistics (ACL).
- Ryan McDonald, Joakim Nivre, Yvonne Quirnbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, et al. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 92–97. Association for Computational Linguistics (ACL).
- Laura A. Michaelis and Hanbing Feng. 2015. What is this, sarcastic syntax? *Constructions and Frames*, 7(2):148–180.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajic, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis M. Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 4034–4043. European Language Resources Association (ELRA).
- Reynier Ortega-Bueno, Francisco Rangel, Delia Irazú Hernández Farias, Paolo Rosso, Manuel Montes-y Gómez, and José E Medina Pagola. 2019. Overview of the task on irony detection in Spanish variants. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019), co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019)*. CEUR-WS.org.
- Juan-Pablo Posadas-Duran, Grigori Sidorov, and Ildar Batyrshin. 2014. Complete syntactic n-grams as style markers for authorship attribution. In *Proceedings of the Mexican International Conference on Artificial Intelligence*, pages 9–17. Springer.
- Ines Rehbein, Josef Ruppenhofer, and Bich-Ngoc Do. 2019. tweeDe – A Universal Dependencies treebank for German tweets. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 100–108. Association for Computational Linguistics (ACL).
- Mickael Rouvier and Pierre-Michel Bousquet. 2017. LIA@DEFT’2017: Multi-view Ensemble of Convolutional Neural Network. In *Actes de l’atelier « Défi Fouille de Textes » (DEFT 2017), 24^{ème} Conférence TALN: Traitement Automatique des Langues Naturelles*, pages 13–26.
- Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, Oronzo Antonelli, and Fabio Tamburini. 2018. PoSTWITA-UD: An Italian Twitter treebank in Universal Dependencies. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA).

- Manuela Sanguinetti, Cristina Bosco, Lauren Cassidy, Özlem Çetinoğlu, Alessandra Teresa Cignarella, Teresa Lynn, Ines Rehbein, Özlem Ruppenhofer, Djamé Seddah, and Amir Zeldes. 2020. Treebanking User-Generated Content: A Proposal for a Unified Representation in Universal Dependencies. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC 2020)*. European Language Resources Association (ELRA).
- Djamé Seddah, Farah Essaidi, Amal Fethi, Matthieu Futral, Benjamin Muller, Pedro Javier Ortiz Suárez, Benoît Sagot, and Abhishek Srivastava. 2020. Building a user-generated content north-african arabizi treebank: Tackling hell. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1139–1150. Association for Computational Linguistics (ACL).
- Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández. 2012. Syntactic dependency-based n-grams as classification features. In *Proceedings of the Mexican International Conference on Artificial Intelligence*, pages 1–11. Springer.
- Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández. 2013. Syntactic dependency-based n-grams: More evidence of usefulness in classification. In *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics*, pages 13–24. Springer.
- Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández. 2014. Syntactic n-grams as machine learning features for natural language processing. *Expert Systems with Applications*, 41(3):853–860.
- Grigori Sidorov. 2014. Should syntactic n-grams contain names of syntactic relations?. *International Journal of Computational Linguistics Applications*, 5(2):25–47.
- Natalia Silveira, Timothy Dozat, Marie-Catherine De Marneffe, Samuel R Bowman, Miriam Connor, John Bauer, and Christopher D Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 2897–2904. European Language Resources Association (ELRA).
- Maria Simi, Cristina Bosco, and Simonetta Montemagni. 2014. Less is More? Towards a Reduced Inventory of Categories for Training a Parser for the Italian Stanford Dependencies. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, page 83–90. European Language Resources Association (ELRA).
- Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642. Association for Computational Linguistics (ACL).
- Dan Sperber and Deirdre Wilson. 1981. Irony and the use-mention distinction. *Philosophy*, 3:143–184.
- Mariona Taulé, Maria Antònia Martí, and Marta Recasens. 2008. AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC 2008)*, pages 96–101. European Language Resources Association (ELRA).
- Akira Utsumi. 1996. A unified theory of irony and its computational formalization. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 2*, pages 962–967. Association for Computational Linguistics (ACL).
- 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 (SemEval 2018)*, pages 39–50. Association for Computational Linguistics (ACL).
- Chuhan Wu, Fangzhao Wu, Sixing Wu, Junxin Liu, Zhigang Yuan, and Yongfeng Huang. 2018. THU_NGN at SemEval-2018 Task 3: Tweet Irony Detection with Densely Connected LSTM and Multi-task Learning. In *Proceedings of The 12th International Workshop on Semantic Evaluation (SemEval 2018)*, pages 51–56. Association for Computational Linguistics (ACL).
- Jaime Zabalza, Jinchang Ren, Jiangbin Zheng, Huimin Zhao, Chunmei Qing, Zhijing Yang, Peijun Du, and Stephen Marshall. 2016. Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. *Neurocomputing*, 185:1–10.