
# Automatic Stance Detection on Political Discourse in Twitter

**Author:** Elena Zotova

**Advisors:** Rodrigo Agerri and German Rigau

Hizkuntzaren Azterketa eta Prozesamendua
Language Analysis and Processing

## Master's Thesis

---

**Department**: Computer Systems and Languages, University of the Basque Country UPV/EHU.

---

Donostia San Sebastián
September 2019

**Acknowledgments**

I would like to acknowledge my warmest thanks to my supervisors Dr Rodrigo Agerri and Dr German Rigau for their friendly and understanding guidance, constructive comments and ideas. I prepared this thesis living in Madrid, and my supervisors made everything to make our remote meetings, discussions and work convenient and easy, and always helped when I had doubts.

Besides, I wish to thank all the professors in the Master of Language Analysis and Processing of the University of the Basque Country. It was a pleasure and a big luck for me to meet people such passionate about linguistics, NLP and their native language. I have learned a lot.

I would also wish to express my gratitude to Manuel Nuñez, head of department of statistic research in Intercom Strategies, Madrid, where I worked during preparing and writing this master thesis, for his expert advice and permission to use the company's software for my research.

Also, I would like to thank Francisco Rangel, from PRHLT Research Center, Universitat Politécnica de Valéncia, one of the organizers of Task on Multimodal Stance Detection in Tweets on Catalan #1Oct Referendum in IberEval 2018, who kindly provided the dataset.

Finally, I would like to say special thanks to my boyfriend for his support, inspiration, listening to multiple stories about this thesis, and participating in rehearsals of the presentation. And thanks to my new friends I have met during my stay in Donostia for fruitful conversations and encouragement.

Thank you all who helped me.

**Abstract**

The majority of opinion mining tasks in natural language processing (NLP) have been focused on sentiment analysis of texts about products and services while there is comparatively less research on automatic detection of political opinion. Almost all previous research work has been done for English, while this thesis is focused on the automatic detection of stance (whether he or she is favorable or not towards important political topic) from Twitter posts in Catalan, Spanish and English. The main objective of this work is to build and compare automatic stance detection systems using supervised both classic machine and deep learning techniques. We also study the influence of text normalization and perform experiments with different methods for word representations such as TF-IDF measures for unigrams, word embeddings, tweet embeddings, and contextual character-based embeddings. We obtain state-of-the-art results in the stance detection task on the IberEval 2018 dataset. Our research shows that text normalization and feature selection is important for the systems with unigram features, and does not affect the performance when working with word vector representations. Classic methods such as unigrams and SVM classifer still outperform deep learning techniques, but seem to be prone to overfitting. The classifiers trained using word vector representations and the neural network models encoded with contextual character-based vectors show greater robustness.

**Keywords**: Text Categorization, Stance Detection, Opinion Mining, Supervised Machine Learning

# Contents

# List of Figures

# List of Tables

# 1 Introduction

The importance of Natural Language Processing (NLP) methods for opinion mining grows due to the rapidly increasing amounts of text information produced in Internet by mass media and users in social media. This text data is unstructured, but contains valuable knowledge that may be useful for various purposes. It gives easy and massive access to public feedback about almost every topic of society: services, products, politics and many others; while the classical methods of public opinion such as polls and surveys are expensive and require much more time and human resources. That is why the demand of automatic categorization and opinion mining artificial intelligence algorithms is ever increasing. This information is of big interest for marketers, researchers, and politicians, but also for business intelligence, government intelligence, decision support systems, political and social researches.

Social media has a great impact in Spain. For example, Spain is one of the top ten most active countries in Twitter. In January 2019 the microblogging service in this country had 6.01 million users[1]. Furthermore, the discourse on this microblogging platform in Spain is highly politicized. Every political event resonates in posts of social media immediately, and in Twitter more promptly than in other media. All Spanish politicians and political activists have their Twitter accounts and directly express opinions on many points, provoking heated debates and generating news reports.

Twitter gives access to new information instantly, allowing to share opinions, facts, pictures, videos, and to participate in discussions. Moreover, the analysis of social media content may reveal some insights. For example, an analysis of about 900.000 tweets (Boynton and Jr., 2016) showed that the public in the United Kingdom had already been in favor of Brexit since 2012, and stayed like that until the referendum, but the traditional opinion polls were not able to detect that. In addition, some studies such as Hill et al. (2013); Bovet et al. (2016) affirm that extracted opinions from social media show a strong correlation to the opinions obtained via traditional approaches such as polls and surveys.

Taking this into account, in this thesis we will focus on the stance detection of political discourse in Twitter for three languages: Catalan, Spanish and English. We will examine the stance detection in debates about the Catalonia self-determination Referendum which took place in Catalonia on the 1st of October 2017. The referendum was approved by the Catalan Parliament on September the 6th but was declared illegal by the Spanish Government and prohibited by the Constitutional Court of Spain a few days after. That provoked heated debates between the supporters ("independentistas") and the opponents ("unionistas").

Twitter is a source of great amounts of data for text classification and opinion mining which would be impossible to process manually by humans. The data is easy to collect as the platform provides free access to its public API[2] (application programming interface) which allows to connect to its databases and receive data in response to specific requests

[1] https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/
[2] https://developer.twitter.com/en/docs

like keyword, period, place etc.

Text classification on social media posts such as tweets is a challenging NLP task. Tweets are very short, up to 240 characters, and there are a lot of messages that contain very few words, specific vocabulary, slang, orthographic mistakes and non-grammatical phrases, emojis, hashtags, images, links, pieces of irony and sarcasm. The same user can write in two or three languages. Twitter message is instant, highly emotional and sometimes difficult to classify even for humans, especially if we only have available a few tweets from some user without context and without connections with other participants of the discussion.

## 1.1 Stance Detection: Problem Statement

Stance is a way of thinking about some topic, problem, person or object, expressed as a public opinion [3]. In Natural Language Processing (NLP), stance detection refers to automatically predicting whether the author of a given statement is in favor of some topic, event or person (target) or against it. Sometimes a text does not expresses any stance towards the target, in this case the stance is neutral, or absent (Ghanem et al., 2018).

More specifically, the problem is the following. We have a set of tweets and a given target. For instance, the referendum on self-determination of Catalonia in 2017. The task is for the NLP to detect whether the text represents the stance in favor of the target topic or against it (or whether none of them are expressed). Below there are some examples in Spanish from the MultiStanceCat dataset (MultiModal Stance Detection in tweets on Catalan #1Oct Referendum) (Taulé et al., 2018).

> Example 1: *Los catalanes que dicen que quieren #democracia (votar en el #referéndum del #1O), no independencia, mienten.*
>
> Translation: *The Catalans who say that they want #democracy (to vote on the #referendum of #1O), but not independence, lie.*
>
> Stance: AGAINST

According to the task definition, a NLP system for stance detection should classify this sentence taking into account only textual information, without any knowledge about the Twitter user, the thread of the conversation and the connection with other users. Example 1 is relatively "easy", because the target topic and the attitude to it is explicit. In many cases the stance is expressed indirectly, without naming the topic but mentioning the persons, facts or events related to the topic, as shown by Example 2, for which we need background information to know that the referendum was held on the 1st of October.

> Example 2. *Me voy a meter en la cama el 30 de septiembre y no voy a salir hasta el 5 o el 6 d octubre #Catalunya #Independencia #1OCT #Hastaelmismisimo*

[3] `https://dictionary.cambridge.org/dictionary/english/stance`

Translation: *I am going to the bed the 30th of September and won't leave it till the 5th or 6th of October #Catalunya #Independencia #1OCT #Hastaelmismisimo*

Stance: AGAINST

There are some posts which, in isolation, are very difficult or just impossible for humans to deduce whether the statement is against or favor to the Catalan referendum, as it can be seen in Example 3. We can see here the lack of evidence of the attitude to the topic, without the specific hashtag we cannot even guess what it is about. Furthermore, a post may not support any of the controversial points of view, and be "against everything", like in Example 4.

Example 3: *De verdad, no hay mas ciego que el que no quiere ver. #Catalunya #1octubreARV*

Translation: *Really, there is no more blind one than one who does not want to see. #1octubreARV*

Example 4. *Vaya desastre de debate. Y vaya politicos que tenemos. Todos. Uff que nivel #1octubreARV*

Translation: *What a mess the debate is. And what politicians we have. All of them. Uff what a level #1octubreARV*

Stance: NEUTRAL

It is considered that stance detection has much in common with the sentiment analysis. In sentiment analysis, systems try to determine whether the polarity of a given text is positive, negative, or neutral. In stance detection, in contrast, systems are to determine if the author says that he/she is against or for some given topic, which may be implicit and difficult to extract from the short text. The difference is that the text may express negative opinion about an entity contained in the text, but one can also conclude that the author is favorable towards the target (a person, an event, an entity or a topic). It means we cannot consider negative sentiment in the message as an equivalent of "against" stance (Sobhani et al., 2016). Thus Example 5 expresses a negative attitude towards Inés Arrimadas, a Spanish politician while at the same time expressing a favour stance towards the target topic (the referendum).

Example 5: *Arrimadassss para de decir tonterías, nos vamos!! #1octL6*

Translation: *Arrimadassss stop saying nonsense, we are going!! #1octL6*

Stance: FAVOR

One more difference between stance and sentiment that stance may be express without using any emotional words. In Example 6 there are not any sentiment words, but it is possible to infer that the author is against of the referendum, because it will cause a big expense for the Catalan government.

Example 6. *#1octl6 Y ... esto quien lo paga?*

Translation: *#1octl6 And ... who will pay for this?*

Stance: AGAINST

Summarizing, automatic stance detection is not a trivial task and present some specific challenges, such as implicit targets, absence of context and complex ways to express stance.

## 1.2 Research Objectives

In this thesis we focus on building multilingual systems for automatic stance detection on political discourse in Twitter. Our research objectives are the following:

1. To build classification models able to detect stance in a Twitter message towards the topic related to politics using machine learning techniques, exploring different approaches to feature selection.

2. To study the influence of text pre-processing and normalization of social media posts for the stance detection task.

3. To study word and document vector representations and its influence on the performance of the classification systems for stance detection.

4. To analyze and compare the performance of the systems with the state of the art.

5. To explore the robustness of the stance detection systems across languages, aiming to develop systems with the ability to generalize across datasets and languages.

6. To analyze the errors of the applied algorithms and propose the ways of improvement of the classification systems.

## 1.3 Document Description

The rest of the thesis is organized as follows. In Section 2 we review several state-of-the-art works related to automatic stance detection. We present the main approaches for Spanish, Catalan and English languages. In Section 3 we describe the methodology applied to our research. We present the datasets from the IberEval 2018 (Taulé et al., 2018) and SemEval 2016 (Mohammad et al., 2016b) shared tasks which are used for training and evaluating the classification models. Also, we describe the algorithms and tools used to undertake our experimentation. In Section 4 we describe the experimental set-up of the work and the different systems built and evaluated. We offer some concluding remarks about the tuning of our systems and provide an error analysis of our system. We finish this master's thesis in Section 6 with the final conclusion and discussion for future work.

# 2 State of the Art

## 2.1 Opinion Mining, Sentiment Analysis and Stance Detection

With the growing availability and popularity of opinion-rich textual resources such as online review sites, personal blogs and social media, new opportunities and challenges arise as people and organizations use natural language technologies to understand the opinions expressed by others. Sentiment Analysis is the sub-field of Natural Language Processing (NLP) that studies people's opinions, sentiments, and attitudes towards products, organizations, entities or topics. Although several complex emotion categorization models have been proposed in the literature (Russell, 1980; Cambria et al., 2010) most of the Sentiment Analysis community has assumed a simpler categorization consisting of two variables: subjectivity and polarity. A text is said to be **subjective** if it conveys an opinion, and objective otherwise. We understand **polarity classification** as the task of telling whether a piece of text (document, sentence, phrase or term) expresses a sentiment.

Automatic stance detection is the task of classifying the attitude expressed in a text towards a given target Mohammad et al. (2016b). Stance detection can be viewed as a subtask of Opinion Mining being also closely related to Sentiment Analysis Pang et al. (2008) and Text Classification Aggarwal and Zhai (2012).

There are three levels in the investigation of sentiment analysis: document level, sentence level and aspect level.

- The task in the document level is to predict negative, positive or neutral sentiment of the whole document such as article, social media post or user review (Pang et al., 2002; Turney, 2002; Dave et al., 2003). This task can be approached using very well known Text Classification techniques. The problem appears when the document contains opinions of various targets.

- Sentence level is usually related to objectivity classification in order to distinguish subjective, objective o neutral information (Wiebe et al., 1999; Finn et al., 2002). Again this task can be addressed applying Text Classification techniques.

- In the aspect level also known as Aspect Based Sentiment Analysis (ABSA) the goal is to determine the sentiment towards different aspects of the topic, product o service rather than the whole text (Pontiki et al., 2014, 2015; Poria et al., 2016; Pablos et al., 2018). For example, in mobile phone reviews, the speed of the processor and the screen resolution may be estimated with different polarity in the same message: "the speed is good but the screen is too poor". ABSA takes into account the different targets of an statement, which makes it similar to stance detection task.

Earlier attempts to computationally assess sentiment in text were based on document classification (Pang et al., 2002; Turney, 2002; Hu and Liu, 2004). As a classification problem, multiple machine learning techniques have been applied to Sentiment Analysis Liu (2012). This task is usually a two-class classification problem (positive vs. negative).

Sometimes a third class (neutral) is introduced. Because many of them relied on the presence of polar words and/or co-occurrence statistics, a document guarantees to have a fair amount of clues. Also, some sources such as movie reviews provided ready-to-use document level annotations, making it possible to develop the first supervised systems (Pang et al., 2002). Supervised machine learning techniques require gold-standard datasets to be collected and annotated manually by humans for being used as training and testing data for the classifiers.

As for any supervised machine learning classification problem, one of the most important task is *f*eature engineering. There are several categories of features that have been tried to represent textual examples for opinion mining.

N-grams with different weighting schemes such as frequency and TF-IDF. This approach is successfully used in the majority of text classification tasks. Pang et al. (2002) was the first who categorized movie reviews into positive and negative and showed that the models trained with unigrams as features and Naïve Bayes and Support Vector Machines (SVM) classifiers performed better that other approaches.

Part-of-speech of the words may provide information, for example, adjectives indicate that the speech is more emotional and subjective.

Lexicon based features are sentiment words and phrases are specific entities for expressing negative or positive sentiments in a text. We know that *good*, *incredible* and *fantastic* usually used to say something positive and *bad*, *awful* and *terrible* are about something negative. Lexical dictionaries and databases such as WordNet Miller et al.; Qiu et al.; Al-Kabi et al.; Baccianella et al.; San Vicente et al. are used for the feature generation.

Syntactic dependencies generated from dependency parsing are also have been used together with classical features. (Xia and Zong, 2010; Ng et al., 2006)

For classification, a wide range of methods is used, among them statistical models, regressions, support vector machines and recently, deep neural networks.

Unsupervised learning includes lexicon-based approaches, which are difficult for social media due to non-grammatical entities and misspelled words (Taboada et al., 2011; Hu et al., 2013). The approach of Lin and He (2009); Pablos et al. (2018); Shams and Baraani-Dastjerdi (2017) is based on combining topic modelling with Latent Dirichlet Allocation (LDA) and sentiment classification.

Sentiment analysis is highly dependent on the domain, and a model trained on specific dataset may perform poorly on the unseen texts from different domain. Also, there is a lack of well labelled datasets for different targets. Lately, in order to resolve this problem, cross domain sentiment analysis is being addressed. The main goal of this task is transfer the knowledge from labelled data to target data where no labelled or a limited corpus exist. In this scenario the researchers study shared features and relations between the domains (Blitzer et al., 2006; Pan et al., 2010; Li et al., 2009). Now, the main approach is deep learning including recurrent neural networks with attention mechanism (Chen et al., 2012; Glorot et al., 2011; Zhang et al., 2019). Supervised deep learning systems use to require large volumes of manually annotated data (Chen et al., 2017; Araque et al., 2017) although very recent unsupervised contextual word embeddings such as BERT Devlin et al. (2019) pre-trained on very large corpora are obtaining state-of-the art results fine tuning

the model with very small training data.

## 2.2 English Language

Stance detection as a subtask of opinion mining and sentiment analysis is a relatively new research area in NLP. From the beginning, the main approach for stance classification has been supervised machine learning. Initial works mainly focused on congressional debates (Thomas et al., 2006) or debates in online forums (Somasundaran and Wiebe, 2009; Murakami and Raymond, 2010; Anand et al., 2011; Walker et al., 2012; Hasan and Ng, 2014; Sridhar et al., 2014). These domains are specific because the gold labels can easily be obtained. For instance, in Sridhar et al. (2014) approach the authors collect posts from various authors from social media debate sites and forums, where all posts are connected as dialogues or responses to the main post of the discussion. The posts are linked to one another by agreement or rebuttal links and are already "labelled" for stance, either *P*RO or *C*ONTRA. Main approaches for debate stance detection have used sentiment and argument lexicons, statistical measures and counts, n-grams, repeated punctuation, part-of-speech tagging, syntactic dependencies and may others (Wang et al., 2019).

Research on Twitter posts started in 2014, and presented a new challenge to the NLP community since tweets are short, informal, full of misspellings, shortenings, slang and emoticons, and are produced in large amount with great velocity. Researchers Rajadesingan and Liu (2014) were the first who determined stance at user level. They assumed that if many users retweet a particular pair of tweets in a short time, then this is likely that this pair of tweets had something in common and share the same opinion on the topic.

The first Stance Detection in Tweets task[4] was presented in 2016 as a part of SemEval challenge organized by the National Research Council Canada. The task aimed to detect stance from single tweets, without taking into account conversational structure of online debates and information about authors. The SemEval competition included two subtasks. Task A was formulated as follows: "given a tweet text and a target entity (person, organization, movement, policy, etc.), automatic natural language systems must determine whether the tweeter is in favor of the given target, against the given target, or whether neither inference is likely" Mohammad et al. (2016b). In Task B the goal was to detect stance in relation of unseen target. The organizers of the challenge also prepared a new dataset which consisted of 4.000 tweets in English corresponding to five stance targets, such as abortions, religion, climate changes, etc., and was annotated both with stance and sentiment labels.

The baseline system designed for the challenge built by the organizers obtained the best results. This baseline system outperformed the submissions from all 19 teams that had participated in the competition. The features used in the system were the following: word and character n-grams, average word embeddings, and sentiment features. A supervised system was trained using a support vector machine classifier (Mohammad et al., 2016b). Other approaches were based on convolutional neural networks (CNN) (Wan Wei,

[4]http://alt.qcri.org/semeval2016/task6/

2016; Vijayaraghavan et al., 2016), recurrent neural network (RNN) (Zarrella and Marsh, 2016), ensemble model (Liu et al., 2016), maximum entropy classifier and domain dictionaries (Krejzl and Steinberger, 2016), etc. Most of the participants used standard text classification features such as word and sentence vector embeddings and n-grams.

The state of the art in stance detection on the SemEval 2016 dataset currently is hierarchical attention based neural model proposed by Sun et al. (2018). In this approach, implemented after the SemEval task, the document is represented under the influence of linguistic features. A linguistic attention mechanism (Kim et al., 2017) is used to learn the correlations between document representation and different linguistic features. The linguistic features are sentimental word sequence, dependency-based features and argument sentence. The document level representation is based on pre-trained word2vec vectors. Documents and linguistic feature sets are encoded with various LSTMs.

In Task B of the SemEval task the stance targets not always present in the message and no training data is available to test the targets. For example, a tweet with positive stance towards Donald Trump is also a negative stance towards Hillary Clinton as implicit target. The best system was proposed by Augenstein et al. (2016). The novel approach is focused on target-dependent representations of tweets. The experiment is done with conditional encoding with neural architecture (LSTM), which builds a representation of the tweet that is dependent on the target. The researchers combine the vector representation on the target and the vector representation of the tweet, and predict stance for target-tweet pair. They also train word2vec model on a large amount of tweets containing targets.

## 2.3 Spanish and Catalan Languages

We should note that all mentioned works were implemented on the English dataset. The first challenge in stance detection in Spanish and Catalan languages was carried out in IberEval 2017[5], the 2nd Workshop on the Evaluation of Human Language Technologies for Iberian languages, during the SEPLN 2017 conference. The organizers of the workshop offered a task related to automatic stance detection and presented a dataset of tweets in Spanish and Catalan (Bosco et al., 2016; Mariona Taule and Pattí, 2017) where the independence of Catalonia is discussed[6]. The authors of the best system of the task do experiments with different types of features such as part of speech, lemmas, hashtags, length of tweets etc., and a set of classifiers (Lai et al., 2017).

In 2018, the third IberEval 2018 workshop[7] co-located with the SEPLN 2018 conference also included a stance detection task. The aim of the MultiModal Stance Detection in tweets on Catalan #1Oct Referendum task at IberEval 2018 (MultiStanceCat) was to detect the authors stances–in favor, against or neutral– with respect to the Catalan October, 1 Referendum (2017) in tweets written in Spanish and Catalan from a multimodal perspective. The dataset also contained images from the given tweets (Taulé et al., 2018).

[5]http://nlp.uned.es/IberEval-2017/index.php/

[6]http://stel.ub.edu/Stance-IberEval2017/data.html

[7]http://www.autoritas.net/MultiStanceCat-IberEval-2018/

The best results on this task were obtained by a team from the Carlos III University (LABDA). They presented a system based on simple bag-of-words approach with TF-IDF vectorization (Segura-Bedmar, 2018). They evaluated several of the most broadly used classifiers. The colleagues do not report anything about pre-processing of the text. They only try to train the model using single tweets with context and without it. This baseline obtained F1 score=0.28 in test evaluation on Spanish dataset.

For the Catalan, the Casacufans team approached the task using texts and images. They used hashing vectorizing from the Scikit-learn toolkit to pre-process and represent texts, and they use linear SVM to train the model. With respect to images, the participants trained a Convolutional Neural Network to detect Spanish or Catalan flags. As report the organizers of the task, the authors did not provide a working note explaining their approach in details (Taulé et al., 2018).

The approach of the Polytechnic University of Valencia team (CriCa) (Cuquerella and Rodríguez, 2018) was to combine datasets of Spanish and Catalan to create a larger corpus and make it more balanced. They did various experiments. The baseline was simple tokenizing and training a Linear SVM classifier. The first model was with stemming with different length of the stem (three, four and five characters) and removing fixed number of characters from the ending of the word. Since Spanish and Catalan share many words, especially, stemming helped to generalize. Additionaly, and some tweets also contain texts in both languages.

It should be underlined that in the Spanish and Catalan datasets, there is still a wide margin to test state of the art NLP approaches to build better stance classifications systems.

# 3 Methodology

For our investigation we need 1) annotated corpora for training supervised systems and 2) tools for building automatic stance recognition systems. The experimentation has been undertaken using the following tools: RapidMiner software and Scikit Learn[8] (Pedregosa et al., 2011) to train SVM supervised machine learning models, Gensim[9] (Řehůřek and Sojka, 2010) for working with word embeddings and Flair[10] (Akbik et al., 2018) as a deep learning platform to train Recursive Neural Networks for stance detection.

## 3.1 Datasets

In the experiments we use three datasets: in Spanish, Catalan and English. The Spanish and Catalan datasets first were presented in IberEval shared task in 2018 (Taulé et al., 2018) and the English dataset was presented within the 2016 SemEval shared task (Mohammad et al., 2016b).

### 3.1.1 TW-1O Referendum corpus, IberEval 2018

The datasets in Spanish and Catalan were collected as follows. The organizers used the #1oct, #1O, #1oct2017 and #1octl6 hashtags to select the tweets to be included in the TW-1O Referendum corpus[11] (Taulé et al., 2018). These hashtags were the most widely used (especially the first two) in the debate on the right to hold a unilateral referendum on Catalan independence from Spain. A total of 87,449 tweets in Catalan and 132,699 tweets in Spanish were collected from September, 20 to the day before the Referendum was held (2017 September, 30). From this data the TW-1O Referendum corpus was built. The final consists of 11,398 tweets: 5,853 written in Catalan (the TW-1OReferendum CA corpus) and 5,545 in Spanish (the TW-1OReferendum ES corpus).

| **Language** | **Train** | **Test** | **Total** |
|---|---|---|---|
| Catalan | 4684 | 1169 | 5853 |
| Spanish | 4437 | 1108 | 5545 |

Table 1: Train and test datasets of TW-1O Referendum corpus.

Each tweet is provided in context, which is formed by its previous and next tweets from the user's Twitter timeline, also including any pictures that the tweets may contain. Eighty percent of the corpus was used for training purposes, while the remaining twenty percent was used for testing. The Spanish part of the corpus is relatively well balanced, as we shown by Figure 2, while the Catalan set is hugely skewed towards the FAVOR class, as illustrated by Figure 1.

[8] `https://scikit-learn.org`
[9] `https://radimrehurek.com/gensim/`
[10] `https://github.com/zalandoresearch/flair`
[11] `http://www.autoritas.net/MultiStanceCat-IberEval2018/corpus/`

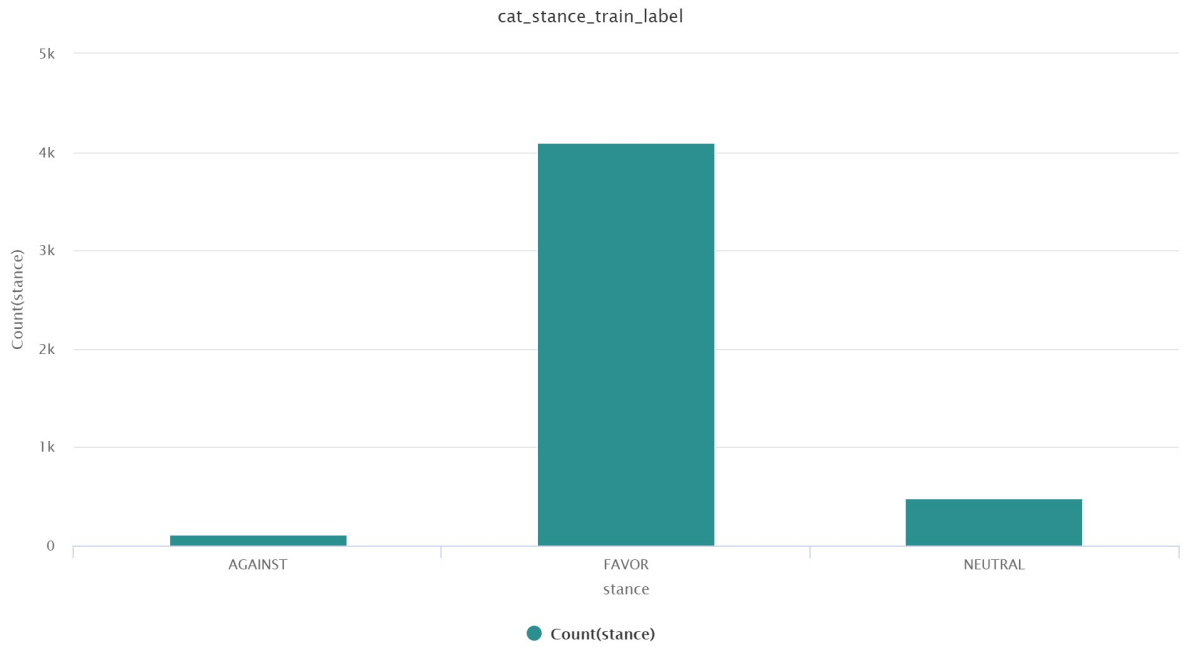


Figure 1: Distribution of classes in the Catalan dataset.

Here we can see some tweet examples from the corpus.

1. Tweet: *Res ni ningú, ens aturarà #Votarem #DretaDecidir #1Oct #CatalunyaLliure #defensemlademocracia http://t.co/PgVLYH8AgN*

Language: Catalan

Stance: FAVOR

Translation: *Nothing and nobody will stop us #Votarem #DretaDecidir #1Oct #CatalunyaLliure #defensemlademocracia http://t.co/PgVLYH8AgN*

2. Tweet: *Mientras tanto en #España se espera una REPRESION para todo público este #1Oct Tan democráticos ellos... https://t.co/gw7QIfrjHk*

Language: Spanish

Stance: FAVOR

Translation: *Meanwhile in #España a REPRESSION is expected by the general public this #1Oct Very democratic them... https://t.co/gw7QIfrjHk*

3. Tweet: *Adeu #1octubreARV #1octubrenovotare http://t.co/x3dXO3v7np*

Language: Catalan

Stance: AGAINST

Translation: *Bye bye #1octubreARV #1octubrenovotare http://t.co/x3dXO3v7np*

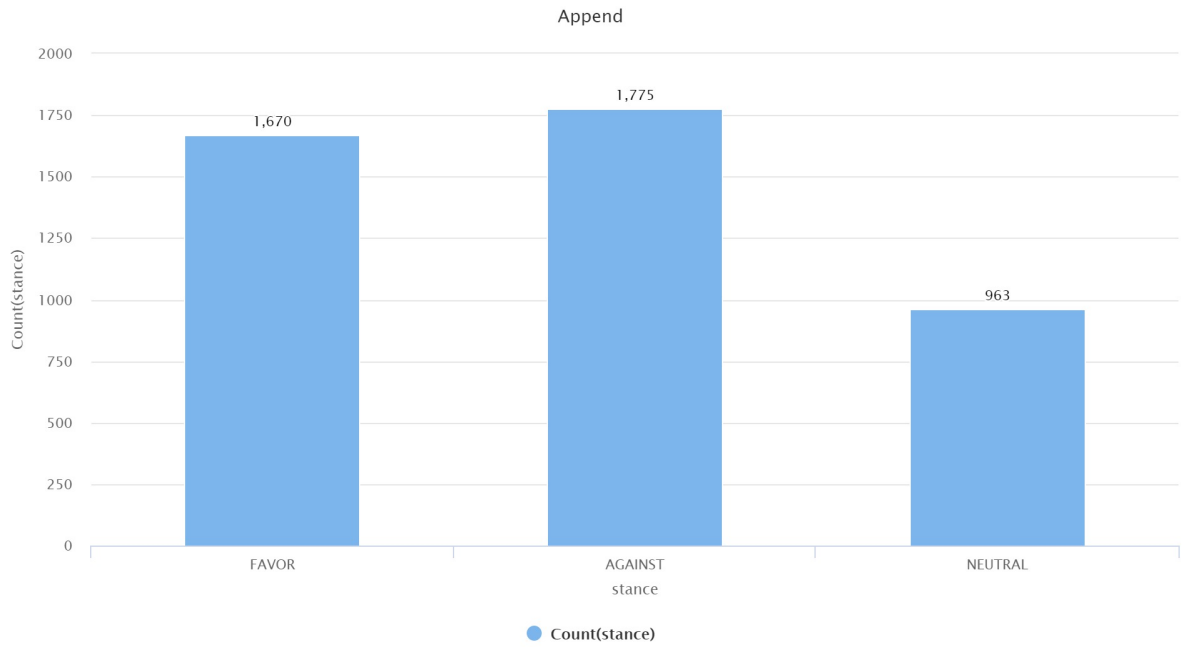


Figure 2: Distribution of classes in the Spanish dataset.

4. Tweet: *Más q votos creo q estais usando personas jugando con sus sentimientos SABIAIS q el #1Oct ES ILEGAL https://t.co/1SJcwn7LHd*

Language: Spanish

Stance: AGAINST

Translation: *You know that more than votes you are using persons playing with their sentiments YOU KNOW that the #1Oct IS ILLEGAL https://t.co/1SJcwn7LHd*

5. Tweet: *Voteu! #1Oct Crees que la respuesta del Estado al desafío independentista catalán está siendo adecuada? https://t.co/LlZrkd20gh vía @20m*

Language: Catalan+Spanish

Stance: NEUTRAL

Translation: *Vote! #1Oct Do you think that the States response to the Catalan pro+independence challenge is appropriate? https://t.co/LlZrkd20gh va @20m*

6. Tweet: *Necesito alguien con quien comentar #1octL6*

Language: Spanish

Stance: NEUTRAL

Translation: *I need someone to comment on #1octL6 with*

### 3.1.2 SemEval 2016

The dataset in English[12] was presented at the Stance Detection task organized at SemEval 2016 (Mohammad et al., 2016b). It consists of tweets labeled for both stance and sentiment (Mohammad et al., 2016a). We do not use the sentiment labels available in this dataset in order to keep the same experimental setup as for the Spanish and Catalan datasets, for which not sentiment annotations are available. In the supervised track, more than 4,000 tweets are annotated with respect to five targets: "Atheism", "Climate Change is a Real Concern", "Feminist Movement", "Hillary Clinton", and "Legalization of Abortion". For each target, the annotated tweets were ordered by their timestamps. The first 70 percent of the tweets formed the training set and the last 30 percent were reserved for the test set.

To prepare the dataset, the organizers collected 2 million tweets containing favor, against and ambiguous hashtags for the selected targets. Each of the tweets was also annotated for whether the target of opinion expressed in the tweet is the same as the given target of interest. The organizers made a small list of query hashtags and split them into three categories: favor, against and ambiguous. Later, the hashtags were removed from the corpus. As only tweets with hashtags in the end of the tweet were used, the grammar and syntactic structure are kept. The authors organized a questionnaire and crowdsourcing setup for annotating stance. Each tweet was annotated by eight respondents (Mohammad et al., 2016a).

| **Target** | **Train** | **Test** |
|---|---|---|
| Feminist Movement | 664 | 285 |
| Hillary Clinton | 639 | 295 |
| Legalization of Abortion | 603 | 280 |
| Atheism | 513 | 220 |
| Climate Change is a Real Concern | 395 | 169 |
| **Total** | **2814** | **1249** |

Table 2: Number of examples per target in the SemEval 2016 English dataset.

Some examples from SemEval 2016 dataset follow.

1. Tweet: *I still remember the days when I prayed God for strength.. then suddenly God gave me difficulties to make me strong. Thank you God! #SemST*

Target: Atheism

Stance: AGAINST

2. Tweet: *@PH4NT4M @MarcusChoOo @CheyenneWYN women. The term is women. Misogynist! #SemST*

Target: Feminist Movement

Stance: FAVOR

[12] `http://alt.qcri.org/semeval2016/task6/data/`

## 3.2 Tools and Algorithms

### 3.2.1 RapidMiner

The first part of our experiments was done with RapidMiner[13], a data science software platform that provides an integrated environment for text mining, machine learning, and predictive analytics. It is suitable for commercial applications as well as for research, education, training, and prototyping. The software supports all steps of the machine learning process including data preparation, results visualization, model validation and optimization (Hofmann Markus, 2013).

RapidMiner provides many algorithms for supervised learning and clustering, including extensions for working with WEKA, Word2Vec, Keras, etc., allowing to build neural networks, preprocessing text (tokenize, apply stopwords etc) and encoding of regular expressions. It allows to process data in various formats, such as CSV, Excel or XML (except JSON). It allows to execute any type of Python or R scripts. It makes the software flexible and suitable for many basic tasks in text mining. RapidMiner is developed on an open core model. The RapidMiner Studio Free Edition is limited to one logical processor and 10.000 data rows.

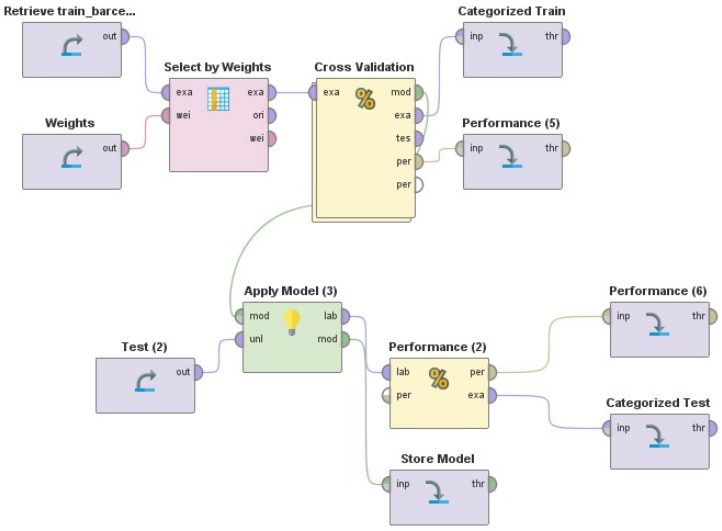


Figure 3: The workflow in RapidMiner: the training and evaluating process.

The main concepts of RapidMiner are:

- *process* – a workflow where a special task is done;
- *operator* – a function, for example the Process documents operator that executes some operations for text processing, or the Cross Validation that executes a cross validation function over a selected estimator. An operator placed into a process may be nested which means that an operator may execute some more subprocesses inside the main process;
- *example set* – data in a specific format of the application. It can be visualized as a table, or graphics, or statistics. One example is a row. Any example set may be exported to CSV, TXT or XLS format;

[13]https://rapidminer.com

- *attribute* – a variable, in the Example set it is a column.

In Figure 3 we provide an example of a process of training and testing a model. Datasets are connected to a classifier and then to the model, the Cross Validation operator is nested, and inside there are some more operators, such as a Classifier and Performance. The output of the process is saved as an example set with new attributes-prediction. It also contains the performance scores and the model stored as objects in RapidMiner format. All the text data can be exported to a table or text files.

### 3.2.2 Word Vector Representations

Vector representations of words, characters or documents, also known as "embeddings" is a technique for creating language models in continuous real valued representations. We use the following types of vector representations in our experiments:

- Static word embeddings: the most well-known are word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), and FastText (Joulin et al., 2017). They are pre-trained over large corpora and are able to capture semantic and syntactic similarities based on co-ocurrences.

- Character-level embeddings (Lample et al., 2016; Ma and Hovy, 2016), which are trained during the learning process on specific data to capture subword features.

- Contextual string embeddings such as Flair (Akbik et al., 2018) which is an embedding for a string of characters in a context.

We use pre-trained FastText and Flair contextual string embeddings as a method of word vector representation. Since Catalan is a less-resourced language, there are not so many NLP resources for its processing. However, FastText provide word vector models trained on Common Crawl[14] and Wikipedia corpora using a Continuous Bag-of-Words (CBOW) architecture with position-weights and 300 dimensions. The Fastext models for Spanish, Catalan and English based on Common Crawl contain around 2 million words each.

The CBOW architecture used in Fasttext was (see Figure 4 first proposed as part of the Word2vec model (Mikolov et al., 2013). Originally there were two architectures implemented in Word2vec: CBOW and skip-gram. CBOW is a bag-of-words model because the order of words in the context does not influence the prediction, but it uses continuous distributed representation of the context. In other words, the CBOW architecture predicts the current word from context words not paying attention to their order. According to the authors, CBOW model is faster then skip-gram. Most importantly, the FastText model is capable of building word vectors for words that do not appear in the vocabulary of the model, i.e. they were not presented in the training set for this model.

[14] `http://commoncrawl.org/`

In order to produce vectors for out-of-vocabulary words, such as proper names and misspelled words, FastText word vectors are built from vectors of substrings of characters which are character n-grams of length 5, a window of size 5 and 10 negatives (Grave et al., 2018). In order to do so, FastText averages the vector representation of its character n-grams. However, it fails in vectorizing totally non-grammatical entities that the model has never seen. It means that the accurate pre-processing of the text is important.

Flair contextual string embeddings are based on neural language modeling that allows language to be modeled as distributions over sequences of characters instead of words. According to the authors, "this type of embeddings is trained without any explicit notion of words and thus fundamentally model words as sequences of characters, and are contextualized by their surrounding text, meaning that the same word will have different embeddings depending on its contextual use" (Akbik et al., 2018). These embeddings are pre-trained on very large unlabeled corpora and are able to capture semantic meaning in context and therefore produce different embeddings for ambiguous words depending on their usage. Since Flair embeddings represent words and context as sequences of characters, it handles better misspelled and rare words, and is able to detect subword structures and morphology. The authors admit that "by learning to predict the next character on the basis of previous characters, such models capture semantic and syntactic properties: even though trained without an explicit notion of word and sentence boundaries, they have been shown to generate grammatically correct text, including words, subclauses, quotes and sentences" (Akbik et al., 2018).

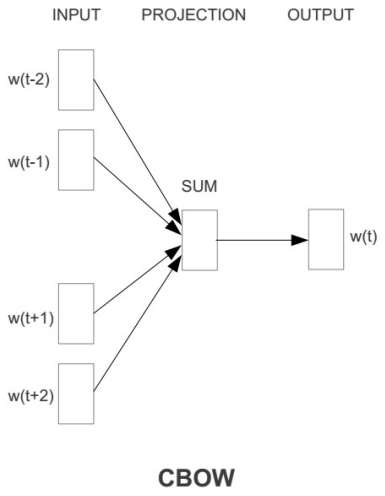


Figure 4: Continuous bag-of-words proposed by Mikolov et al. (2013).

### 3.2.3 Classifier: Support Vector Machines

For the experiments with classic machine learning algorithms we selected Support Vector Machines (SVM) classifier (Cortes and Vapnik, 1995) which is implemented in both the RapidMiner and scikit-learn toolkits. SVM is a non-probabilistic linear classifier. The main advantage of SVM method is that it is robust to high dimensionality and sparsity of the feature vectors. It was proven that it is one of the most efficient learning methods for text classification because texts usually have a large number of features and not many of them are totally irrelevant. Also, text classification problems are linearly separable (Joachims, 1998).

SVM tries to find a hyperplane, which is a separating line, between data of two (or more) classes (The scheme is in the Figure 5). Then the algorithm finds a support vector in such a way that its points are the closest to the line dividing the two classes. The

distance between the hyperplane and the support vectors is computed; this is functional margin.

The hyperplane that has the largest distance to the nearest training-data point of any class separates the data in the best way. The larger the margin, the lower is the generalization error of the classifier (Hastie et al., 2001). With the appropriate kernel function, SVMs can be used to learn radial basics function (RBF) networks, polynomial classifiers, and three-layer sigmoid neural nets (Joachims, 1998).

The SVM learner has various hyper-parameters to be set, such as kernel type, SVM type, C, gamma, and some more. Hyper-parameters are those which cannot be obtained during the learning process and should be set before it. We select C-SVC type with RBF kernel type. C-SVC type handles multi-class classification according to a one-vs-one scheme. RBF kernel performs well in the models where relation between class labels and features is nonlinear. In the experiment of Joachims (1998), RBF kernel shows slightly better performance than polynomial.

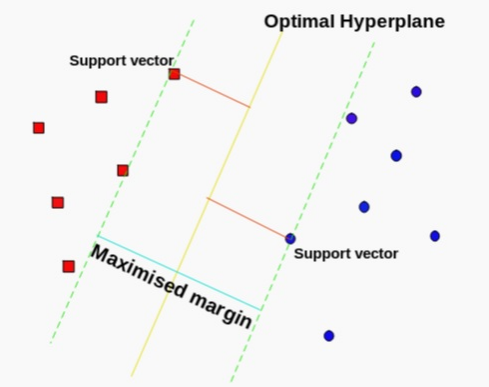


Figure 5: The hyperplane of SVM.

Two most important parameters which impact the accuracy of the model are gamma and C. As explained in the scikit-learn documentation[15], the gamma parameter defines how far the influence of a single training example reaches, with low values meaning far and high values meaning close. The gamma parameters are the inverse of the radius of influence of samples selected by the model as support vectors. The C parameter is used to balance correct classification of training examples against maximization of the decision functions margin. For larger values of C, a smaller margin will be accepted if the decision function is better at classifying all training points correctly. A lower C will encourage a larger margin, therefore a simpler decision function, but the training accuracy will be lower. In other words, C behaves as a regularization parameter in the SVM. Setting a too large C parameter may result in overfitting.

### 3.2.4 Neural Networks and Flair library

Neural networks and contextual character embeddings are currently obtaining best results in many Natural Language Processing (NLP) tasks. A new framework, presented in late 2018, Flair library [16] provides an environment for building classification models based on neural networks. It is a Python library which allows to apply NLP models to many tasks, such as named entity recognition (NER), part-of-speech tagging (PoS), word sense disambiguation, and text classification. In sequence labelling tasks, Flair obtained best results for a number of public benchmarks such as PoS tagging and NER (Akbik et al., 2018). It is multilingual system and contains models for some less-resourced languages,

[15] `https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html`
[16] `https://github.com/zalandoresearch/flair`

such as Catalan. Flair has simple interfaces for combining different word and document embeddings. The framework is based directly on Pytorch, making it easy to train the models and experiment with new approaches. The library allows to prepare the text corpus, calculate the vector representations and build statistical models with recurrent neural networks.

Flair models allow to combine different types of embeddings by concatenating each embedding vector to form the final word vectors in a stack. For instance, stacked embeddings may mix FastText static word embeddings and Flair contextual character embeddings, or Flair with ELMo contextual embeddings (Peters et al., 2018). According to experiments done by the authors of Flair, in many configurations it may be beneficial to combine the Flair embeddings with static word embeddings to add potentially greater latent word-level semantics.

Flair architecture for text classification is based on a BiLSTM which is a type of recurrent neural network (RNN) (Schuster et al., 1997). RNNs are capable of using internal states, so-called memory, to process sequences of input, building the network on all previously seen inputs, which allows to take into account the context of a sentence. Moreover, the value of each hidden layer unit also depends on its previous state, so the word order and the character order are also considered.

The LSTM architecture (Hochreiter and Schmidhuber, 1997) is widely used in NLP tasks and shows state-of-the-art results. However, to achieve this, it requires the selection and optimization of many hyper-parameters (Reimers and Gurevych, 2017). Flair includes a wrapper for tuning the neural network with the hyper-parameter selection tool Hyperopt[17] (Bergstra et al., 2013). Basically, it is a grid search over the hyper-parameters of the neural net and the number of combinations grows exponentially with the number of parameters set. This step is the most time consuming experimentation stage, and in our case can occupy up to 20-30 hours depending on the computing power of the machine and the number of combinations and it is recommendable to execute it on GPU accelerated hardware. We tune the following hyper-parameters:

- Hidden size: the size of LSTM hidden states.
- Dropout set in range 0.0-0.5: a parameter that prevents overfitting for neural networks (Srivastava et al., 2014).
- RNN layers: 1 or 2 layers.
- RNN type: The type of activation function is Rectified Linear Unit (ReLU) (Nair and Hinton, 2010). The activation function is necessary to normalize the output values of the network and make them statistically balanced. The ReLU function is linear for all positive values, and zero for all negative values.
- Learning rate set in range: 0.05, 0.1, 0.15, 0.2. The deep neural network is updated by stochastic gradient descent (SGD) and the parameters (weights) are updated like

[17] `https://github.com/hyperopt/hyperopt`

this: weight = existing weight - learning rate * gradient. The gradient takes into account a loss function that aims to measure inconsistency between the true label and predicted label. When the loss decreases, the accuracy of the prediction increases. Too small a learning rate will make a training algorithm converge slowly while too large a learning rate will make the training algorithm diverge (Bengio, 2012).

- Mini batch size: 16, 32 examples. Mini batch is a subset of training data to calculate the SGD.
- Max epochs: Epoch is one run of the neural network through the entire training set. Set to 100.
- Number of evaluations: 100.

Also, we trained our models selecting a stacked embedding configuration consisting of FastText Common Crawl and Flair contextual character embeddings.

## 3.3 Evaluation Metrics

There are various metrics for the evaluation of performance of the classification models: accuracy, precision, recall and F1 scores. Accuracy is the proportion of correctly classified documents with respect to the total number of documents. We do not use the accuracy score as an evaluation method because in unbalanced datasets the accuracy may be very high. However, this does not show that the classification model performs well.

Precision is the fraction of the correct documents (true positive) among the documents identified as positive (true positive and false positive).

$$Precision = \frac{tp}{tp + tn}$$

Recall is the percentage of the correct documents (true positive) among all the real positive documents (true positive and false negative).

$$Recall = \frac{tp}{tp + fn}$$

F1 score is the harmonic mean of precision and recall. The formula of F1 is the following:

$$F1 = 2\frac{Precision \times Recall}{Precision + Recall}$$

For evaluating models in the cross-validation step, we use F1 macro-average for all three classes. F1 score macro-average is calculated as F1 metric for each label and their unweighted mean value. This does not take imbalanced data into account. In multi-class classification, macro-average F1 treats all classes equally while F1 micro-average favours the more populated classes (Sokolova and Lapalme, 2009).

The organizers of Semeval 2016 and IberEval 2018 used the macro-average score of two classes: F1 score (FAVOR) and F1 score (AGAINST), as the bottom-line evaluation metric. The F1 score of NEUTRAL (NONE) class is not taken into account. They provided a Perl script[18] that calculate the final F-score with this formula:

$$F1_{avg} = \frac{F1_{favor} + F1_{against}}{2}$$

The same metric to evaluate the performance when testing of the experiments is applied.

[18] `http://alt.qcri.org/semeval2016/task6/data/uploads/eval_semeval16_task6_v2.zip`

# 4 Experiments

Our main goal is to experiment on detecting stance in Spanish and Catalan datasets, and then investigate the robustness of the various models by applying them to the English dataset. The experimental work was organized in three different setups:

1. TF-IDF vectorization with a SVM classifier.
2. Vector representation of tweets with FastText models and SVM classifier.
3. Stacked Flair and FastText vector representation of words to train a neural network (BiLSTM).

## 4.1 Pre-processing and Normalization

The first step is to perform text pre-processing. Since each tweet in the Catalan and Spanish data is given in context, with the previous and the next tweet, we use them to enrich the text information. We concatenate all three tweets to one document, so the number of words increases and the weight of each token in a tweet is more representative.

We try to perform exhaustive text normalization. Normalization is a process of putting words to standard forms, which reduces irrelevant and noisy information. In our case it helps to reduce the number of features for TF-IDF feature representation and to raise the number of words that correspond with the vocabulary of pre-trained models. First, we clean the text from punctuation, remove mentions starting with "@", "RT", URLs and numbers. All the words of the corpus were converted to lowercase.

Another part of text normalization is lemmatization, the task of determining that two words have the same root, despite their differences. For example, the Spanish words *voy*, *iba* and *iré* are forms of the verb *ir* (to go). One Spanish verb has 55 forms, including gerunds and participles, and they may be totally different in spelling. Lemmatization maps all forms to the same initial form–infinitive for verbs or singular masculine form for nouns and adjectives–also called dictionary form. This process is essential for languages such as Spanish and Catalan but not so important for English (Jurafsky and Martin, 2018). The most sophisticated methods of lemmatization use full morphological analysis and POS tagging. In our system, we simplified the task up to replacing the word form with its lemma without analysis. A simple Python function uses a dictionary[19] with word forms (values) and their lemmas (keys). It takes each word of a phrase, checks if it is in the values of the dictionary and replaces it with the key of the dictionary. If a word is not in the dictionary, for example if it is a named entity or incorrectly spelled word, the program does not perform any lemmatization.

This method is not accurate, and it is not capable to resolve ambiguities. This means that, for instance, the preposition *para* (for) and the verb *para* (to stop) will be mapped to the same lemma, namely, *parar* (stop). Also it can not handle named entities, especially

[19] `https://github.com/michmech/lemmatization-lists`

Spanish and Catalan surnames, for example, *Ada Colau* is converted to *ada colar*. To reduce the error rate, we manually edited the list of lemmas, and deleted the less frequent ambiguous words. In any case, our experiments showed that this kind of lemmatization reduces dramatically the number of features without loss of semantic information and helps in general to improve results. In addition, it allows to deal with some unseen words. For example, if the word *yendo* (going) does not appear in the training corpus but another form of its lemma does, then both words will be recognized as having the same lemma, namely, the Spanish verb *ir* (to go).

Tokenization is the task of segmenting the text into instances such as characters, words, phrases. We split the tweets by white space which allows to keep all the words with hyphens, and other special symbols. Next, stopwords (auxiliary verbs, prepositions, articles, pronouns and the most frequent words) and words shorter than three characters are removed, usually this kind of words do not contain any relevant information.

The next step is normalization of orthography, which is simple replacing repeated letters with one, all vocals and the most frequent consonants, such as *s, z, h, m, j* etc, and replacing common shortened words to normal form. We do not touch letters that can be double in Spanish and Catalan: *t, l, r*. The result is like following: *holaaaaaaa* is converted to *hola*, *q* is converted to *que*. We also replace all letters with diacritics with simple ones to reduce the number of tokens in the TF-IDF way of vectorization. The replacement is done with regular expressions.

## 4.2 TF-IDF+SVM

The first experimental setup uses a TF-IDF (Term Frequency times Inverse Document Frequency) (Jones, 1972) representation of features extracted from the corpus. This approach is similar to bag-of-words where instead of word frequency the TF-IDF measure is used.

TF-IDF weighting scheme is broadly used for document and text classification, information retrieval and topic modelling. The goal of using of TF-IDF measure is to reduce the impact of words that occur too frequently in a given corpus as they are less informative than features that occur in a small part of the training corpus. The TF-IDF is the product of two statistic metrics, term frequency and inverse document frequency. It is computed as follows:

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

Where:
*tfi,j*= number of occurences of i in j,
*dfi* = number of documents containing i,
*N*=total number of documents

*Term frequency* is the number of times term appears in a document divided on total number of terms in the document (Luhn, 1957). *Inverse document frequency* represents the weight of the less frequent words in all documents. Thus, IDF is log scaled relation between the total number of documents and the number of documents with a given term. It gives a

higher weight to words that are not that frequent in the documents. The idea is that terms that occur too frequently in every document are not as helpful for this task because it does not allow to detect the most meaningful words present in the documents. The more rarely a word occurs in the corpus, the higher it is its IDF score (Jurafsky and Martin, 2018).

We calculate the TF-IDF scores for all unigrams in the training corpus and obtain a document-feature matrix with 13,488 features in Spanish corpus, 11,882 features and Catalan corpus and 1,249 for the English data. The number of features equals the size of the vocabulary of the dataset and represents the dimensionality of the document vector. It is extremely sparse, since the documents contains about 20-30 words. So, our task is to reduce the dimensionality of the vector.

### 4.2.1 Feature Selection

Before training the model we decide what kind of features could be most useful. The goal of the feature selection process is to identify the most relevant features and discard redundant information. Irrelevant features provide almost zero useful information about the dataset, these can be words that occurs in all classes with the same frequency. Feature selection may affect significantly the performance of the classification model reducing overfitting, training time and improving accuracy.

We use the Information Gain method for feature selection, which is a term from information theory (Cover and Thomas, 2006). In machine learning, this measure provides a way to calculate the mutual information between the features and the classification classes. According to Aggarwal (2012), mutual information "is defined on the basis of the level of co-occurrence between the class and word". In other words, it represents the predictive power of each feature, and measures the number of bits of information obtained for prediction of a class in terms of presence or absence of a feature in a document. It is a filter method that selects features by ranking them with correlation coefficients (Guyon Isabelle, 2003). The information gain scores show how common is the feature in a target class, for example the words that occurs mainly in tweets labelled as FAVOR stance and almost never in AGAINST stance, are very informative and ranked highly.

We understand that in text classifications the most irrelevant features such as stopwords and short words removed mostly before the creating the document-feature matrix. Nevertheless, our experiments show that there Information Gain produces a small improvement of the performance. We give more details in Subsection 4.2.2, Table 5. All the weights are normalized and all the features ranked from one to zero. We then select those features that are scored larger than zero.

### 4.2.2 Spanish, Catalan and English models

We train a classification model via 10-fold cross validation on the training set using the LibSVM learner (Fan et al., 2005) implemented in RapidMiner. The LibSVM learner allows to apply SVM algorithm for a multiclass problem.

Since the hyper-parameters of the SVM classifier cannot be directly learnt from the

data within estimators, they must be selected before the training process starts. In order to do this, we perform hyper-parameter optimisation by grid-search. The goal of grid-search is to find the optimal hyper-parameters of a model. It is an exhaustive search through a manually specified subset of parameters of a classifier. Exhaustive search is an algorithm that examines all possible combinations of parameters and checks if they fulfil the condition. During the grid-search SVM models with different parameters are built and estimated by performance metrics, accuracy, precision, and recall and measured by 5-fold cross-validation on the training set.

To reduce the cost of the grid-search process, we select two parameters of SVM classifier only, C and gamma. In total, there were 121 combination of the parameters. The following final hyper-parameters were chosen after grid-search. For Spanish: C=700, gamma=0.3; for Catalan: C=100, gamma=0,4507. The kernel is RBF.

After selecting the best parameters we estimate the classification models via 10-fold cross-validation. For the English data, we train five models according to the target and calculate the average for each metric. We obtain the following results for the best models selected according to the cross validation F1 scores (Table 3, Table 4):

| **TF-IDF+SVM** | | **Precision** | **Recall** | **F1** |
|---|---|---|---|---|
| Spanish | AGAINST | 0.75 | 0.76 | 0.76 |
| | FAVOR | 0.73 | 0.75 | 0.74 |
| | NEUTRAL | 0.83 | 0.77 | 0.80 |
| Catalan | AGAINST | 0.71 | 0.17 | 0.27 |
| | FAVOR | 0.96 | 0.99 | 0.98 |
| | NEUTRAL | 0.84 | 0.76 | 0.79 |
| English | AGAINST | 0.69 | 0.79 | 0.66 |
| | FAVOR | 0.69 | 0.39 | 0.42 |
| | NEUTRAL | 0.54 | 0.23 | 0.30 |

Table 3: Cross-validation results of TF-IDF+SVM models

| **Dataset** | **F1 2 classes** | **F1 3 classes** |
|---|---|---|
| Spanish | 0.75 | 0.77 |
| Catalan | 0.62 | 0.68 |
| English | 0.54 | 0.46 |

Table 4: F1 scores for best TF-IDF+SVM models in cross-validation.

The worst performance is for the English dataset. We think that this might due to the small amount of training examples. For the Catalan model the result is predictably worse because of the imbalanced dataset where the AGAINST class is very scarcely represented.

We also trained the Spanish models in order to perform an ablation test to check which components of our system have a bigger impact in performance.

Our experiments show (Tables 5 and 6) that the grid-search optimisation of hyper-parameters strongly affects the accuracy of the model. Training the SVM model with default parameters (C=1, gamma=0), the F1 score dropped dramatically to 0.29. The other parts of the system do not impact the result so strongly, but the accuracy of the model still deteriorates.

| **Spanish Dataset** | **Class** | **Precision** | **Recall** | **F1** |
|---|---|---|---|---|
| Lemmas+Feature selection | AGAINST | 0.75 | 0.76 | 0.76 |
| | FAVOR | 0.73 | 0.75 | 0.74 |
| | NEUTRAL | 0.83 | 0.77 | 0.80 |
| Without grid search | AGAINST | 0.40 | 1 | 0.57 |
| | FAVOR | 0 | 0 | 0 |
| | NEUTRAL | 0 | 0 | 0 |
| Without pre-processing | AGAINST | 0.54 | 0.93 | 0.69 |
| | FAVOR | 0.73 | 0.51 | 0.60 |
| | NEUTRAL | 0.97 | 0.20 | 0.34 |
| Without feature selection | AGAINST | 0.70 | 0.73 | 0.72 |
| | FAVOR | 0.66 | 0.75 | 0.70 |
| | NEUTRAL | 0.66 | 0.46 | 0.54 |

Table 5: Comparison of models for Spanish dataset.

| **Model for Spanish Dataset** | **F1 2 classes** | **F1 3 classes** |
|---|---|---|
| Lemmas+Feature selection | **0.75** | **0.77** |
| Without grid search | 0.29 | 0.19 |
| Without preprocessing | 0.65 | 0.54 |
| Without feature selection | 0.71 | 0.65 |

Table 6: F1 scores for models trained with different parameters evaluated on 10-fold cross-valdation.

We can conclude that text pre-processing and proper feature selection is crucial for good results in the stance detection tasks. Reducing the number of features helps to generalize and minimize the cost in time of training and predicting.

### 4.2.3 Catalan and Spanish Combined

We performed an additional experiment training on a mixed dataset of Spanish and Catalan examples. Catalan and Spanish languages are grammatically similar and share some lexical entities, after lemmatization in particular, so we suppose that it may help to generalize better. We search the best parameters and pre-processed the text in the same way as we did for the Spanish and Catalan systems in the previous section.

The corpora is pre-processed separately for each language. We do lemmatization, remove mentions, "RT", links, punctuation, stopwords (dictionary of stopwords consists of both Spanish and Catalan words), words shorter than three characters, tokenize, lowercase, perform text normalization and then concatenate corpora in one dataset. In total there are 9,098 tweets, and the class distribution is the following: AGAINST=1,895, FAVOR=5,761, NEUTRAL=1,442. The test dataset is mixed, as well, and consists of 2,277 examples.

After pre-processing, TF-IDF vectorization is applied, and the output of the process is a document-feature matrix with 26,947 unigrams as features. The model is trained with following the procedure detailed in the previous section (with grid-search over C and gamma parameters and 121 combinations, from which C=1, and gamma=1.8 are obtained). After testing, we obtained a F-score of 0.6817, which is the best result we obtain for the Catalan language. The cross validation results are in Table 7.

| **System** | | **Precision** | **Recall** | **F1** |
|---|---|---|---|---|
| CAT+SPA | AGAINST | 0.87 | 0.38 | 0.53 |
| | FAVOR | 0.73 | 0.99 | 0.84 |
| | NEUTRAL | 0.95 | 0.32 | 0.47 |
| | F1 avg (2 class) | | | 0.69 |
| | F1 avg (3 class) | | | 0.62 |

Table 7: Performance of Catalan+Spanish model in cross-validation.

## 4.3 FastText+SVM

In this part is explained the second approach where we use word vector representations for the tweets, or tweet embeddings as features and SVM classifier.

As we know, the word vector dimension in FastText model is 300. Multiplied by the number of words in the corpus that we extract as features, we would obtain a too complex matrix to process by the SVM classifier. To reduce the dimensionality, we propose to represent a tweet as an average value of the vectors of the words from a given tweet. We are aware of that this method has some limitations. It discards the word order and sentence semantics. However, according to previous work, "averaging the embeddings of words in a sentence has proven to be a surprisingly successful and efficient way of obtaining sentence embeddings" (Kenter et al., 2016). A similar approach has been used in various tasks such as sentiment analysis (Júnior et al., 2017; Socher et al., 2013) and sentence2vec representation (Ben-Lhachemi and Nfaoui, 2018; Pagliardini et al., 2018). The average tweet vector representation is calculated as following:

$$V(t) = \frac{1}{n}\sum_{i=1}^{n} W_i$$

where
*V(t)* is a vector of a tweet,

$n$ is a number of words in a tweet,

$W$ is a word vector.

The FastText model is loaded using the Gensim library (Řehůřek and Sojka, 2010). To calculate a tweet vector, we coded a simple function which first assigns a vector to each token in a tweet, skipping non-processable entities, and calculate the average vector for the tweet. It returns a document-feature matrix with 300 features.

Since the words in the vocabulary of the model are from real world texts, Wikipedia and news, they are well spelled, hence we change the text pre-processing method: we do not replace diacritics and stopwords. We remove punctuation, numbers, hashtags, mentions, links and RTs as well.

We train two models: with normalized text and full forms with the same classifier as in the previous sections, namely, C-SVM with RBF kernel, C=10, gamma=1 for Spanish and C=100, gamma=0.5 for Catalan selected after grid-search optimization method. For the English data, we train five models, one for each target, and average the evaluation results. The results are shown in Tables 8 and 9.

| System | | Precision | Recall | F1 |
|---|---|---|---|---|
| Lemmatized SPA | AGAINST | 0.55 | 0.77 | 0.65 |
| | FAVOR | 0.61 | 0.54 | 0.57 |
| | NEUTRAL | 0.60 | 0.29 | 0.39 |
| Not lemmatized SPA | AGAINST | 0.57 | 0.69 | 0.63 |
| | FAVOR | 0.60 | 0.57 | 0.59 |
| | NEUTRAL | 0.50 | 0.36 | 0.42 |
| Lemmatized CAT | AGAINST | 0 | 0 | 0 |
| | FAVOR | 0.91 | 0.99 | 0.95 |
| | NEUTRAL | 0.91 | 0.34 | 0.50 |
| Not lemmatized CAT | AGAINST | 0 | 0 | 0 |
| | FAVOR | 0.91 | 0.99 | 0.95 |
| | NEUTRAL | 0.81 | 0.35 | 0.49 |
| Not lemmatized ENG | AGAINST | 0.50 | 0.94 | 0.66 |
| | FAVOR | 0.78 | 0.03 | 0.06 |
| | NEUTRAL | 0.53 | 0.20 | 0.29 |

Table 8: Cross validation results for FastText+SVM models.

The results show that there is a very small difference in F1 score between pre-processed and raw text for the Spanish and Catalan experiments. We can hypothesize that reducing of the number of word forms in the text is not important for the word vector representation. As the Spanish model trained on non-lemmatized text performs 0.01 better in F1 macro, we consider it to be the best configuration from the cross-validation results.

| System | F1 2 classes | F1 macro |
|---|---|---|
| Lemmatized SPA | 0.61 | 0.54 |
| Not lemmatized SPA | 0.61 | 0.55 |
| Lemmatized CAT | 0.48 | 0.48 |
| Not lemmatized CAT | 0.48 | 0.48 |
| Not lemmatized ENG | 0.36 | 0.34 |

Table 9: F1 scores in cross-validation for FastText+SVM system.

## 4.4 Neural Architecture

In the third setup we use the neural architecture implemented in Flair library described in Subsection 3.2.4. The classification model combines FastText embeddings and Flair embeddings to represent the text. Furthermore, the Flair system implements a bidirectional Long short-term memory (BiLSTM) architecture.

The train corpus is split into train and development part in a 0.9/0.1 proportion. The optimisation value is F1 macro, and the performance of the model during training will be estimated on the development set. We optimize parameters on the development set to avoid any overfitting and then train the model with the best parameters selected during the grid-search. Like in the previous experiments, two types of corpora, lemmatized and raw, were used. The text was cleaned from punctuation, numbers, mentions, hashtags and links.

The table 10 demonstrates that the text with full word forms is categorized better than the lemmatized one. Taking this into account, in the text pre-processing we can skip some steps and make it less time consuming. It is enough to correct some orthography and remove punctuation and numbers.

| FLAIR | F1 macro |
|---|---|
| Lemmatized SPA | 0.4926 |
| Not Lemmatized SPA | 0.5832 |
| Lemmatized CAT | 0.5017 |
| Not Lemmatized CAT | 0.5840 |
| Not Lemmatized ENG | 0.4210 |

Table 10: The F1 score of systems trained with Flair architecture.

We apply the best combination on the English dataset and see that the result on English dataset is significantly worse. As mentioned earlier, this might be due to the fact that neural models require larger datasets to obtain competitive performances.

## 4.5 Overall Training Results

The best results obtained via cross validation for three languages are given in Table 11:

| System | SPA | CAT | ENG |
|---|---|---|---|
| Best TF-IDF+SVM | **0.75** | **0.68** | **0.46** |
| Best FastText+SVM | 0.61 | 0.48 | 0.34 |
| Best Flair | 0.58 | 0.58 | 0.42 |
| Best IberEval 2018/ SemEval 2016 | 0.73 | no data | no data |

Table 11: F1 scores of best models on the training set.

We compare the best systems and conclude that for all datasets the TF-IDF+SVM system obtains the highest F1 scores outperforming the Spanish best system in the IberEval 2018 shared task. To achieve these results it was necessary to pre-preprocess the text data (lemmatize, normalize, clean and remove ) and make feature selection in order to reduce the dimensionality of the document-feature space.

In FastText+SVM and Flair systems where vector representation of words and documents are used, the best configuration is done without text lemmatization which makes the pre-processing step faster.

## 4.6 Test Evaluation

We selected the best models for each language according to the cross-validation and development scores and evaluated them on the gold standard test datasets. The organizers of both challenges, SemEval 2016 and IberEval 2018, use the same type of the evaluation metric, the macro-average of the F1 score of two classes, FAVOR and AGAINST. We evaluate our systems with the same metric and provide the results in Table 12 for Spanish in Table 13 for Catalan.

| System | F1 2 classes |
|---|---|
| TF-IDF+SVM+Feature selection+Lemmatization | **0.6078** |
| TF-IDF+SVM+Cat+Spa | 0.5900 |
| FastText+SVM | 0.5827 |
| Flair+Stacked embeddings+RNN | 0.5598 |
| Best IberEval TF-IDF+SVM (Segura-Bedmar, 2018) | 0.2802 |

Table 12: Test performance on TW-1O Referendum corpus for Spanish language.

The final evaluation demonstrates that all proposed systems in Spanish and Catalan outperform the IberEval 2018 best official test results. The best system of IBEREVAL 2018 task for Spanish dataset, presented by Segura-Bedmar (2018), is a similar to our TF-IDF+SVM system, but more simple. They used TF-IDF measure over unigrams, combined the tweet with the target of the stance with its context (previous and next tweet), and did grid-search over SVM hyper-parameters. We believe that our system obtains better results because we pay more attention to text normalization, including lemmatization. Furthermore, we also perform feature selection in order to reduce noise,

| System | F1 2 classes |
|---|---|
| TF-IDF+SVM+Feature selection+Lemmatization | **0.5844** |
| TF-IDF+SVM+Cat+Spa | 0.5620 |
| FastText+SVM+Tweet embeddings | 0.4996 |
| Flair+Stacked embedding+BiLSTM | 0.5582 |
| Best IberEval SVM+stemming (Cuquerella and Rodríguez, 2018) | 0.3068 |

Table 13: Test performance on the TW-1O Referendum corpus for Catalan language.

as well as hyper-parameter tuning. We also think that the systems we developed based on word embeddings for the representations of tweets (FastText+SVM and Flair+Stacked embeddings+BiLSTM) are more robust than previous work. Thus, the best IBEREVAL 2018 system obtains a F1 score of 0.73 in the cross-validation phase but its performance degrades to 0.28 when evaluated on the test set, which shows that the model is overfitting to the training set.

The system of Cuquerella and Rodríguez (2018) combine both Spanish and Catalan corpora. They also apply stemming and train the model with a Linear SVC classifier. The mixed Cat+Spa+SVM model trained on both Catalan and Spanish dataset is evaluated on Spanish and Catalan test sets separately. As for Spanish, we believe that our models obtain better scores for Catalan due to the work done on normalization and also because the word embeddings representations work better than the systems presented at IBEREVAL 2018. In the following we give an example (Example 4.6) of how our best model, TF-IDF+SVM, works.

> System: TF-IDF+SVM SPA
>
> Main tweet: *#EspanaSaleALaCalle @Alternativa VOX @CiudadanosCs @PPopular @PSOE #1octL6 ¿Pregunten porque los bastardos no reclaman el Rosellón francés*
>
> Next tweet: *RT @Miotroyo2parte: En 2014 falleció un hombre por la caída de un árbol en Madr i gobernaba Ana Botella. Hoy, con @ManuelaCarmena al frent...*
>
> Previous tweet: None
>
> Pre-processed text: preguntar bastardo reclamar rosellon frances fallecer hombre caer arbol madrid gobernar botella frent
>
> Stance: AGAINST
>
> Predicted: AGAINST

The performance model for English dataset is significantly lower than the state-of-the-art results on the SemEval data (See Table 14). We get the best F1 score in our experiments in the model which was trained with the FastText pre-trained word embeddings and the SVM classifier. The English SVM classifiers (both TF-IDF and FastText features) is

| System | F1 2 classes |
|---|---|
| TF-IDF+SVM (proposed) | 0.51 |
| FastText+SVM (proposed) | 0.56 |
| Flair+Stacked embedding+BiLSTM (proposed) | 0.34 |
| Best SemEval 2016 RNN+pre-trained embeddings (Zarrella and Marsh, 2016) | 0.68 |
| Benchmark SemEval 2016 Word embeddings+SVM (Mohammad et al., 2016b) | **0.69** |
| NN+hierarchical attention (Sun et al., 2018) | 0.61 |

Table 14: Comparison of test performance on SemEval 2016 Stance Detection dataset for English language.

comparable to the results for Spanish. The performance of Flair models are significantly lower, just as it was at the cross-validation stage.

We explain the poor performance with the following reasons. All the previous systems for English dataset are rather sophisticated. For instance, Zarrella and Marsh (2016) improve the system with additional large dataset and the system consists of two RNN classifiers that predict task-relevant hashtags on a very large unlabeled Twitter corpus and classify stance. The benchmark model of SemEval 2016 organizers is implemented with sentiment feature in addition to stance label. Sun et al. (2018) apply the most recent approach in neural networks, hierarchical attention neural model, and use both linguistic and structural features. Our approach by comparison is simple and may be improved in various ways.

# 5 Error Analysis

In order to understand better the difficulties of building multilingual stance detection systems we performed a manual error analysis by extracting the wrong predictions for the Spanish and Catalan testsets. Table 15 shows the absolute number of errors for each system.

| **System** | **SPA: 1108 test examples** | **CAT: 1169 examples** |
|---|---|---|
| TF-IDF+SVM | 503 | 145 |
| FastText+SVM | 503 | 139 |
| Flair | **540** | **155** |

Table 15: Number of incorrectly predicted examples in test datasets.

To analyse the errors we did the following:

1. We first calculated the confusion matrices of the systems' predictions.
2. We compared the wrong predictions of the systems.
3. Finally, a manual examination of 100 misclassified examples was performed in order to categorize the various types of errors.

1. The confusion matrices detailing the distribution of errors can be seen in Appendix A, Tables 18 and 19. We can conclude that the most common errors for the Spanish models are due to the true AGAINST class being predicted as FAVOR. In Catalan, AGAINST is often predicted as FAVOR or NEUTRAL because the algorithm is highly biased towards the majority class (FAVOR), and the minority class (AGAINST) is misclassified.
2. The comparison of the misclassified examples (Table 16) shows that more than 50 per cent (in respect to each system) of wrong predictions for all three models in Spanish intersect. The TF-IDF and FastText models share the greatest amount of errors. 185 examples in Spanish and 18 examples in Catalan are common for all three models, which means that none of the systems is able to classify them correctly.

| Combination of systems | SPA Errors in common | CAT Errors in common |
|---|---|---|
| TF-IDF - FastText | **330** | **101** |
| FastText - Flair | 281 | 24 |
| Flair - TF-IDF | 257 | 21 |
| ALL | 185 | 18 |

Table 16: Number of errors common for the systems.

3. We categorized the wrong predictions to see what kind of information impacts more the accuracy of the classifiers. For this type of analysis, we took 100 randomly sampled

misclassified examples from the set which is common for all three systems in Spanish and 100 examples from Catalan. We examined each document and we came with some explanations as to why the system would commit an error. We found out six categories different reasons which are mostly related to pre-processing and the quality of the text data. Note that a single example may occur more one category (Table 17).

| **Error type** | **SPA** | **CAT** |
|---|---|---|
| Code switching | 15 | 23 |
| Different topics in context | 26 | 25 |
| Short document | 21 | 9 |
| Pre-processing | 26 | 13 |
| Non-grammatical tokens | 15 | 4 |
| Bias | 21 | 35 |

Table 17: Error Types over 100 examples

The categories of the errors are described below.

**Code switching**

Code-switching occurs when a speaker shifts from one language to another, so two or three different languages are mixed in the same example, such as Spanish and English, or Spanish and Catalan. This problem is typical for the countries where a part of the citizens are bilingual, and Catalonia is one of the bilingual regions of Spain.

> Example 1. True NEUTRAL - predicted AGAINST in all systems.
>
> #1octL6 no os dais cuenta que no hay solucion? Ahi esta sentado los dos tipos de catalanes que hay en el pueblo y estan fracturados...—Was out mountain biking 8.29 km with #Endomondo. See it here: http://t.co/b6BkZVWiof—#1octL6 La cara de Iceta es un poema...ciudadanos y pp peleandose...
>
> Example 2. True NEUTRAL - predicted FAVOR in all systems.
>
> Passi el que passi l' #1O aquests darrers dies s'han destruït tants ponts entre ESP i CAT que res tornarà a ser com abans—RT @jesusespinosa: Es de Jordi Cotrina. Es la mejor foto de la semana. Los padres del niño de Rubí de 3 años muerto en La Rambla consue...—RT @jonathanmartinz: MUY FAN de @tvetve. https://t.co/jeFwYn3hAJ

**Pre-processing errors**

Social media are noisy texts which are not entirely cleaned by the pre-processing step. We used regular expressions remove hashtags and usernames but the variability is too great

to be completely exhaustive. Furthermore, and as it was previously said, the lemmatizer is not 100 percent accurate.

> True NEUTRAL - predicted AGAINST and FAVOR.
>
> Original text: Dice Albiol que no hay más independentismo ni crisis, pues nada, negando la realidad va a lograr mucho diálogo. #1octL6——El tema es que con toda la decadencia que se ha sufrido en España, esto pase por la independencia. #Vergüenza #CatalanReferendum #10ctARV
>
> Preprocessed text: Dice Albiol que no hay más independentismo ni crisis pues nada negando la realidad va a lograr mucho diálogo El tema es que con toda la decadencia que se ha sufrido en España esto pase por la independencia *üenza*

**Different topics in context**

The concatenation of contextual tweets may add some noise. The main tweet may be about the target topic but a the previous or next tweets could be about other topics.

> True AGAINST - predicted FAVOR.
>
> *@cesc4official me dice un madridista que te vuelvas al Arsenal para poder ganar al Barça* — #1octL6 el problema es que no se leen libros, la historia se repite.— RT @Cazatalentos: Arriba: el pueblo de Madrid pidiendo dignidad Abajo: fascistas apretando filas y cantando el Cara al Sol https://t.co/Gzx...

**Non-grammatical tokens**

This type of errors refers to spelling errors, slang and emotional entities, and errors from the corpus compilation stage. For example, the Twitter database shortens some documents according to the number of characters and a sentence may be end abruptly.

> Example 1. True NEUTRAL - predicted AGAINST.
>
> #1octL6 Si en cada debate, diálogo... se tiran a la yugular, ¿se puede esperar mañana un desenlace pacífico? Pinta muy mal.—@marca 10—#1octL6 Como nos dio un toque @anapastor, vamos a bajar el tono y darnos de hostias igual, pero con cariño. *Enga...*

**Short documents**

Some tweets are too short and lacking in context so that they do not offer any linguistic cues to help the classification task. Also, the pre-processing stage may shorten the tweets to just a few words.

Example 1. True NEUTRAL - predicted FAVOR.

Original text: Pl Espanya en aquest moment! #1O https://t.co/btcvL9oRMy——

Preprocessed text: Pl Espanya en aquest moment

**Biased data**

Given that many of the incorrectly categorized tweets are well spelled and pre-processed, we can assume that the classifier is being misled but some specific linguistic expressions in the documents. Thus, in the Catalan dataset the reason is mainly the poor distribution of the classes, which causes for the majority of the examples to misclassified as FAVOR.

For the Spanish data we examined the training sets and extracted the most frequent words for each class (see Table 23 in Appendix A). For instance, the word *España* and *Cataluña* occur more times for the AGAINST class whereas *Catalunya* and *votar* are likely to appear more often for the FAVOR class.

Example 1. True NEUTRAL - predicted FAVOR in all systems.

Albiol en la tele...100 independentistas por minuto #1octL6—Que previsibles son los cambios de la selección #RioRTVE—@InesArrimadas Déjate de bromitas #Lodijoalbiol #1octL6

Example 2. True FAVOR - predicted AGAINST.

Que dice Arrimadas que no planteeis la posiblidad de modificar la CE para una Cataluña independiente porque a ella no le gusta #1octL6——

**Improvements**:

As a conclusion, for future improvements in detecting stance we would need to improve a number of issues. As the performance of the models depends strongly on the quality of the training data, we should revisit the dataset preparation. While the classes in the Spanish data are balanced, the Catalan dataset is no skewed that no amount of class weighting can correct the strong bias towards the FAVOR class. Furthermore, after error analysis it can be seen that the pre-processing and normalization steps could be improved. Finally, perhaps using language identifying systems and cross-lingual word embeddings may be helpful in processing code-switched language.

# 6 Conclusions

In this thesis we have done a series of experiments for automatic stance detection in social media. We have implemented systems for Spanish, Catalan and English implementing mechanisms for text normalization, different learning features (unigrams over TF-IDF vectorization, static and dynamic word embeddings, both at word and character level) for two different classification methods: one based on SVM and the other one on a BiLSTM neural network. All of our Spanish and Catalan models outperform previous work on the 1Oct Referendum corpus, including the official IBEREVAL 2018 results.

Our experiments show that text pre-processing benefits the performance for models trained on unigrams with SVM (TF-IDF). Lemmatization, as part of the normalization and reduction of dimensionality of the feature-document matrix is shown to be crucial. For the models using continuous word representations (FastText+SVM and Flair) lemmatization is not required given that the word embedding models are computed from word forms. Thus, for those models the pre-processing is minimal (punctuation and numbers are removed).

Feature selection is also performed to further reduce the sparsity of the feature-document matrix. We use information gain, a statistic metric, to estimate which features are important for class prediction. It does not impact the performance as much as the pre-processing and SVM tuning of hyperparameters, but it can be used to make the training process much faster without accuracy loss.

For the TF-IDF+SVM model, reducing the amount of features helps but the model is not that robust in relation of unseen data. This can be observed by comparing the cross-validation and test results, since the variance between the cross validation F1 performance and test F1 performance is quite significative (See Tables 20, 21, 22). As oppose to this, the model using word embeddings for document representation (FastText+SVM and Flair) are much stable and the performance on the unseen test data is closer to the results obtained in the cross-validation stage. Overall, the FastText+SVM system seems to be the most robust across the three languages, despite the difference in the quality of the datasets.

SVM models are not time and resource consuming and may be implemented on relatively noisy data. At the same time, the results obtained are still competitive with respect to newer deep learning systems. In this sense, it seems that neural architectures require larger training data to achieve better results.

## 6.1 Main Contributions

We proposed a set of simple but good performing stance detection systems and compared different methods of text pre-processing and training algorithms for Spanish, Catalan and English languages. The are three main approaches for the development of our systems. A classical approach with TF-IDF features and a SVM classifier, word embeddings for document representation and a SVM classifier, and a stacked vector feature representation to train a neural network architecture. The systems based on vector representation of words and (FastText+SVM and Flair) behave more robustly on unseen test data, and their development is language agnostic.

We have compared various corpus pre-processing methods and provide a detailed error analysis of their performance. As we have seen, the systems share a large proportion of errors. Furthermore, we have shown that text normalization is most effective for the TF-IDF approach.

The systems are ready to be used for real-world tasks and easy to implement. Evaluated on the TW-1O Referendum corpus (Taulé et al., 2018), the systems obtain the best results published up to date for the Catalan and Spanish data.

The source code and the algorithms are shared and can be used for learning purposes, reproducing the experiments, and further improvement [20].

## 6.2 Future Work

We envisage several ways in which to improve the Stance detection systems.

**1. Feature engineering**

N-grams (bigrams and trigrams) may improve the performance in TF-IDF+SVM model and make it detect the inner structure of text. Additional features such as sentiment vocabulary, part-of-speech, syntactic structure in a tweet, and clusters may also help in the classification problem.

**2. Distant supervision**

The performance of the classification models depends on the quality and the size of the training data. Manual data labelling is extremely expensive and time consuming, that is why it is the bottleneck in building automatic NLP systems. We believe that applying distant supervision may help to obtain larger training data. Furthermore, we plan to leverage cross lingual embeddings models so that we can linguistic information across languages (Bergmanis et al., 2017; Ratner et al., 2017; Artetxe et al., 2018).

**3. Transformers**

The Attention-based approach is a state-of-the-art technique that is currently obtaining very competitive results for a number of NLP tasks (Vaswani et al., 2017; Devlin et al., 2019; Dai et al., 2019; Peters et al., 2018). Attention mechanism is able to learn contextual relations between words, taking into account all text surroundings.

[20] `https://github.com/ZotovaElena/Stance-Detection-in-Twitter`

# References

Charu C Aggarwal and ChengXiang Zhai. A survey of text classification algorithms. In *Mining text data*, pages 163–222. Springer, 2012.

Zhai ChengXiang Aggarwal, Charu C. *Mining Text Data*. Kluwer Academic Publishers, Boston/Dordrecht/London, 2012.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA, 2018.

Mohammed N. Al-Kabi, Amal H. Gigieh, Izzat M. Alsmadi, Heider A. Wahsheh, and Mohamad M. Haidar. Opinion mining and analysis for arabic language. *International Journal of Advanced Computer Science and Applications*, 5(5), 2014.

Pranav Anand, Marilyn Walker, Rob Abbott, Jean E. Fox Tree, Robeson Bowmani, and Michael Minor. Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2Nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, pages 1–9, Stroudsburg, PA, USA, 2011.

Oscar Araque, Ignacio Corcuera-Platas, J. Fernando Sánchez-Rada, and Carlos A. Iglesias. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications*, 77(1):236–246, 2017.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 789–798, 2018.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas, 2016.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *in Proc. of LREC*, 2010.

Nada Ben-Lhachemi and El Habib Nfaoui. Using tweets embeddings for hashtag recommendation in twitter. *Procedia Comput. Sci.*, 127(C):7–15, 2018.

Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade - Second Edition*, pages 437–478. 2012.

Toms Bergmanis, Katharina Kann, Hinrich Schütze, and Sharon Goldwater. Training data augmentation for low-resource morphological inflection. In *Proceedings of the CoNLL*

*SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 31–39, Vancouver, 2017.

J. Bergstra, D. Yamins, and D. D. Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, pages I–115–I–123, 2013.

John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, Stroudsburg, PA, USA, 2006.

Cristina Bosco, Mirko Lai, Viviana Patti, Francisco M. Rangel, and Paolo Rosso. Tweeting in the debate about catalan elections. In *In: Proc. LREC workshop on Emotion and Sentiment Analysis Workshop (ESA), LREC-2016*, pages 67–70, Portorož, Slovenia, 2016.

Alexandre Bovet, Flaviano Morone, and Hernán A. Makse. Predicting election trends with twitter: Hillary clinton versus donald trump. *CoRR*, abs/1610.01587, 2016.

G. R. Boynton and Glenn W. Richardson Jr. Agenda setting in the twenty-first century. *New Media & Society*, 18(9):1916–1934, 2016.

Erik Cambria, Robert Speer, Catherine Havasi, and Amir Hussain. Senticnet: A publicly available semantic resource for opinion mining. In *AAAI fall symposium: commonsense knowledge*, volume 10, 2010.

Minmin Chen, Zhixiang Eddie Xu, Kilian Q. Weinberger, and Fei Sha. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*, 2012.

Tao Chen, Ruifeng Xu, Yulan He, and Xuan Wang. Improving sentiment analysis via sentence type classification using bilstm-crf and cnn. *Expert Systems with Applications*, 72:221–230, 2017.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3): 273–297, 1995.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, New York, NY, USA, 2006.

Carlos Almendros Cuquerella and Cristóbal Cervantes Rodríguez. Crica team: Multimodal stance detection in tweets on catalan 1oct referendum (multistancecat). In *IberEval@SEPLN*, 2018.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2978–2988, 2019.

Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th International Conference on World Wide Web*, pages 519–528, New York, NY, USA, 2003.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

Rong-En Fan, Pai-Hsuen Chen, and Chih-Jen Lin. Working set selection using second order information for training support vector machines. *J. Mach. Learn. Res.*, 6:1889–1918, 2005.

Aidan Finn, Nicholas Kushmerick, and Barry Smyth. Genre classification and domain transfer for information filtering. In *ECIR*, 2002.

Bilal Ghanem, Paolo Rosso, and Francisco Rangel. Stance detection in fake news a combined feature representation. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 66–71, Brussels, Belgium, 2018.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pages 315–323, Fort Lauderdale, FL, USA, 2011.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018.*, 2018.

Elisseeff André Guyon Isabelle. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3(4):1157–1182, 2003.

Kazi Saidul Hasan and Vincent Ng. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 751–762, Doha, Qatar, 2014.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York Inc., New York, NY, USA, 2001.

Craig A. Hill, Elizabeth Dean, and Joe Murphy. *Social Media, Sociality, and Survey Research*. Wiley Publishing, 2013.

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

Klinkenberg Ralf Hofmann Markus. *RapidMiner: Data Mining Use Cases and Business Analytics Applications (Chapman and Hall, CRC Data Mining and Knowledge Discovery Series)*. CRC Press, Florida, USA, 2013.

Minqing Hu and Bing Liu. Mining opinion features in customer reviews. In *AAAI'04: Proceedings of the 19th national conference on Artifical intelligence*, pages 755–760, San Jose, California, 2004. ISBN 0-262-51183-5.

Xia Hu, Jiliang Tang, Huiji Gao, and Huan Liu. Unsupervised sentiment analysis with emotional signals. In *Proceedings of the 22Nd International Conference on World Wide Web*, pages 607–618, New York, NY, USA, 2013.

Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Machine Learning: ECML-98*, pages 137–142, Berlin, Heidelberg, 1998.

Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain, 2017.

Edilson Anselmo Corrêa Júnior, Vanessa Queiroz Marinho, and Leandro Borges dos Santos. NILC-USP at semeval-2017 task 4: A multi-view ensemble for twitter sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, pages 611–615, 2017.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing (3d Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2018.

Tom Kenter, Alexey Borisov, and Maarten de Rijke. Siamese CBOW: optimizing word embeddings for sentence representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016.

Yoon Kim, Carl Denton, Luong Hoang, and Alexander M. Rush. Structured attention networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Peter Krejzl and Josef Steinberger. UWB at SemEval-2016 task 6: Stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 408–412, San Diego, California, 2016.

Mirko Lai, Alessandra Teresa Cignarella, and Delia Irazú Hernández Farías. itacos at ibereval2017: Detecting stance in catalan and spanish tweets. In *IberEval@SEPLN*, 2017.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, 2016.

Tao Li, Vikas Sindhwani, Chris H. Q. Ding, and Yi Zhang. Knowledge transformation for cross-domain sentiment classification. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009*, pages 716–717, 2009.

Chenghua Lin and Yulan He. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 375–384, New York, NY, USA, 2009.

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

Can Liu, Wen Li, Bradford Demarest, Yue Chen, Sara Couture, Daniel Dakota, Nikita Haduong, Noah Kaufman, Andrew Lamont, Manan Pancholi, Kenneth Steimel, and Sandra Kübler. Iucl at semeval-2016 task 6: An ensemble model for stance detection in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 394–400, San Diego, California, 2016.

H. P. Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM J. Res. Dev.*, 1(4):309–317, 1957.

Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany, 2016.

Francisco Rangel Paolo Rosso Cristina Bosco Mariona Taule, M. Antonia Martí and Viviana Pattí. Overview of the task on stance and gender detection in tweets on catalan independence ibereval 2017. In *Proceedings of the Second Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2017)*, pages 158–177, Murcia, Spain, 2017.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *Computation and Language (cs.CL)*, 2013.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. Introduction to WordNet: An On-line Lexical Database*. *International Journal of Lexicography*, 3(4):235–244, 1990.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiao-Dan Zhu, and Colin Cherry. A dataset for detecting stance in tweets. In *LREC*, pages 3945–3952, 2016a.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California, 2016b.

Akiko Murakami and Rudy Raymond. Support or oppose? classifying positions in online debates from reply activities and opinion expressions. In *Coling 2010: Posters*, pages 869–875, Beijing, China, 2010.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 807–814, USA, 2010.

Vincent Ng, Sajib Dasgupta, and S. M. Niaz Arifin. Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 611–618, Sydney, Australia, 2006.

Aitor García Pablos, Montse Cuadros, and German Rigau. W2VLDA: almost unsupervised system for aspect based sentiment analysis. *Expert Syst. Appl.*, 91:127–137, 2018.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 528–540, 2018.

Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th International Conference on World Wide Web*, pages 751–760, New York, NY, USA, 2010.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002.

Bo Pang, Lillian Lee, et al. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135, 2008.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237, 2018.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Ion Androutsopoulos, and Suresh Manandhar. Proceedings of the 8th international workshop on semantic evaluation (semeval 2014), pages 27–35,dublin, ireland, august 23-24, 2014.semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, page 27–35, Dublin, Ireland, 2014.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado, 2015.

Soujanya Poria, Erik Cambria, and Alexander Gelbukh. Aspect extraction for opinion mining with a deep convolutional neural network. *Know.-Based Syst.*, 108(C):42–49, 2016.

Likun Qiu, Weishi Zhang, Changjian Hu, and Kai Zhao. Selc: A self-supervised model for sentiment classification. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 929–936, New York, NY, USA, 2009.

Ashwin Rajadesingan and Huan Liu. Identifying users with opposing opinions in twitter debates. In *Social Computing, Behavioral-Cultural Modeling and Prediction - 7th International Conference, SBP 2014, Washington, DC, USA, April 1-4, 2014. Proceedings*, pages 153–160, 2014.

Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason Alan Fries, Sen Wu, and Christopher Ré. Snorkel: Rapid training data creation with weak supervision. *PVLDB*, 11(3):269–282, 2017.

Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, 2010.

Nils Reimers and Iryna Gurevych. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *CoRR*, abs/1707.06799, 2017.

James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

Inaki San Vicente, Rodrigo Agerri, and German Rigau. Simple, robust and (almost) unsupervised generation of polarity lexicons for multiple languages. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 88–97, 2014.

Mike Schuster, Kuldip K. Paliwal, and A. General. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45:2673–2681, 1997.

Isabel Segura-Bedmar. Labda's early steps toward multimodal stance detection. In *Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018) co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018), Sevilla, Spain, September 18th, 2018.*, volume 2150, pages 180–186, 2018.

Mohammadreza Shams and Ahmad Baraani-Dastjerdi. Enriched lda (elda): Combination of latent dirichlet allocation with word co-occurrence analysis for aspect extraction. *Expert Systems with Applications*, 80, 2017.

Parinaz Sobhani, Saif Mohammad, and Svetlana Kiritchenko. Detecting stance in tweets and analyzing its interaction with sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 159–169, Berlin, Germany, 2016.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, 2013.

Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manage.*, 45(4):427–437, 2009.

Swapna Somasundaran and Janyce Wiebe. Recognizing stances in online debates. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 226–234, Suntec, Singapore, 2009.

Dhanya Sridhar, Lise Getoor, and Marilyn Walker. Collective stance classification of posts in online debate forums. In *Proceedings of the Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, pages 109–117, Baltimore, Maryland, 2014.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

Qingying Sun, Zhongqing Wang, Qiaoming Zhu, and Guodong Zhou. Stance detection with hierarchical attention network. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2399–2409, Santa Fe, New Mexico, USA, 2018.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307, 2011.

M. Taulé, F. Rangel, M. A. Martí, and P. Rosso. Overview of the task on multimodal stance detection in tweets on catalan 1oct referendum. In *IberEval 2018. CEUR Workshop Proceedings. CEUR-WS.org*, pages 149–166, Sevilla, Spain, 2018.

Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of EMNLP*, pages 327–335, 2006.

Peter D. Turney. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 417–424, Stroudsburg, PA, USA, 2002.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

Prashanth Vijayaraghavan, Ivan Sysoev, Soroush Vosoughi, and Deb Roy. Deepstance at semeval-2016 task 6: Detecting stance in tweets using character and word-level cnns. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 413–419, 2016.

Marilyn Walker, Pranav Anand, Rob Abbott, and Ricky Grant. Stance classification using dialogic properties of persuasion. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 592–596, Montréal, Canada, 2012.

Xuqin Liu Wei Chen Tengjiao Wang Wan Wei, Xiao Zhang. pkudblab at semeval-2016 task 6 : A specific convolutional neuralnetwork system for effective stance detection. In *Proceedings of SemEval-2016*, page 384–388, San Diego, California, 2016.

R. Wang, D. Zhou, M. Jiang, J. Si, and Y. Yang. A survey on opinion mining: From stance to product aspect. *IEEE Access*, 7:41101–41124, 2019.

Janyce M. Wiebe, Rebecca F. Bruce, and Thomas P. O'Hara. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 246–253, Stroudsburg, PA, USA, 1999.

Rui Xia and Chengqing Zong. Exploring the use of word relation features for sentiment classification. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1336–1344, Stroudsburg, PA, USA, 2010.

Guido Zarrella and Amy Marsh. MITRE at semeval-2016 task 6: Transfer learning for stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 458–463, 2016.

Kai Zhang, Hefu Zhang, Qi Liu, Hongke Zhao, Hengshu Zhu, and Enhong Chen. Interactive attention transfer network for cross-domain sentiment classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:5773–5780, 2019.

# Appendix A

| System | true AGAINST | true FAVOR | true NEUTRAL |
|---|---|---|---|
| **TF-IDF SPA** | | | |
| pred. AGAINST | 307 | 56 | 83 |
| pred. FAVOR | **113** | 221 | 85 |
| pred. NEUTRAL | 100 | 66 | 77 |
| **FASTTEXT+SVM** | | | |
| pred AGAINST | 332 | 81 | 33 |
| pred FAVOR | **176** | 214 | 28 |
| pred NEUTRAL | 133 | 59 | 51 |
| **FLAIR** | | | |
| pred AGAINST | 293 | 99 | 54 |
| pred FAVOR | **147** | 213 | 58 |
| pred NEUTRAL | 103 | 79 | 61 |

Table 18: Confusion matrix on Spanish test set.

| System | true AGAINST | true FAVOR | true NEUTRAL |
|---|---|---|---|
| **TF-IDF CAT** | | | |
| pred. AGAINST | 4 | 19 | 6 |
| pred. FAVOR | 1 | 975 | 45 |
| pred. NEUTRAL | 0 | **74** | 45 |
| **FASTTEXT+SVM** | | | |
| pred AGAINST | 1 | 28 | 0 |
| pred FAVOR | 2 | 1007 | 9 |
| pred NEUTRAL | 0 | **97** | 22 |
| **FLAIR** | | | |
| pred AGAINST | 4 | 24 | 1 |
| pred FAVOR | 9 | 977 | 32 |
| pred NEUTRAL | 0 | **89** | 30 |

Table 19: Confusion matrix on Catalan test set.

| System SPA | | Precision | Recall | F1 Test | F1 CV |
|---|---|---|---|---|---|
| TF-IDF+SVM Normalized | AGAINST | 0.60 | 0.69 | 0.64 | |
| | FAVOR | 0.64 | 0.53 | 0.58 | |
| | F1 avg | | | **0.61** | 0.75 |
| FastText+SVM | AGAINST | 0.52 | 0.74 | 0.62 | |
| | FAVOR | 0.60 | 0.51 | 0.55 | |
| | F1 avg | | | 0.58 | 0.61 |
| Flair | AGAINST | 0.54 | 0.66 | 0.59 | |
| | FAVOR | 0.55 | 0.51 | 0.53 | |
| | F1 avg | | | 0.56 | 0.58 |

Table 20: Test performance of the Spanish systems in terms of F1 macro score for 2 classes: AGAINST and FAVOR.

| System CAT | Class | Precision | Recall | F1 Test | F1 CV |
|---|---|---|---|---|---|
| TF-IDF+SVM Normalized | AGAINST | 0.80 | 0.14 | 0.24 | |
| | FAVOR | 0.91 | 0.96 | 0.93 | |
| | F1 avg | | | **0.58** | 0.63 |
| FastText+SVM | AGAINST | 0.33 | 0.04 | 0.06 | |
| | FAVOR | 0.89 | 0.98 | 0.94 | |
| | F1 avg | | | 0.50 | 0.48 |
| Flair | AGAINST | 0.31 | 0.14 | 0.19 | |
| | FAVOR | 0.90 | 0.96 | 0.93 | |
| | F1 avg | | | 0.56 | 0.58 |

Table 21: Test performance of the Catalan systems in terms of F1 macro score for 2 classes: AGAINST and FAVOR.

| System ENG | | Precision | Recall | F1 Test | F1 CV |
|---|---|---|---|---|---|
| TF-IDF +SVM Normalized | AGAINST | 0.56 | 0.67 | 0.61 | |
| | FAVOR | 0.44 | 0.41 | 0.40 | |
| | F1 avg | | | 0.51 | 0.54 |
| FastText+SVM | AGAINST | 0.67 | 0.69 | 0.68 | |
| | FAVOR | 0.49 | 0.41 | 0.44 | |
| | F1 avg | | | **0.56** | 0.36 |
| Flair | AGAINST | 0.58 | 0.458 | 0.48 | |
| | FAVOR | 0.25 | 0.282 | 0.24 | |
| | F1 avg | | | 0.36 | 0.36 |

Table 22: Test performance of the English systems in terms of F1 macro score for 2 classes: AGAINST and FAVOR..

| Feature | Total | FAVOR | AGAINST | NEUTRAL |
|---|---|---|---|---|
| españa | 451 | 146 | **244** | 61 |
| cataluña | 384 | 121 | **198** | 65 |
| catalanes | 341 | 120 | 155 | 66 |
| votar | 316 | 206 | 101 | 53 |
| catalunya | 288 | **157** | 53 | 78 |
| democracia | 267 | 117 | 126 | 24 |
| contra | 244 | 93 | 69 | 82 |
| gente | 227 | 88 | 108 | 31 |
| albiol | 206 | 134 | 39 | 33 |
| estado | 197 | 55 | 101 | 41 |
| independencia | 180 | 61 | 98 | 21 |
| ahora | 178 | 74 | 73 | 31 |
| independentistas | 168 | 32 | **123** | 13 |
| hacer | 161 | 46 | 88 | 27 |
| gobierno | 160 | 50 | 69 | 41 |
| años | 138 | 40 | 71 | 27 |
| arrimadas | 136 | **82** | 33 | 21 |
| españoles | 130 | 47 | 70 | 13 |
| gran | 128 | 52 | 57 | 19 |
| bien | 126 | 42 | 60 | 24 |
| favor | 122 | 28 | 53 | 41 |
| catalana | 121 | 40 | 45 | 36 |
| catalán | 121 | 31 | 65 | 25 |
| iceta | 117 | 49 | 48 | 20 |
| hablar | 116 | 38 | 59 | 19 |
| gracias | 108 | 53 | 45 | 10 |
| debate | 107 | 30 | 47 | 30 |
| cara | 106 | 48 | 34 | 24 |
| diálogo | 103 | 16 | 71 | 16 |
| barcelona | 101 | 53 | 24 | 24 |
| civil | 101 | 31 | 31 | 39 |

Table 23: Most frequent words from Spanish dataset that occur more than 100 times.