Using Lexical Resources for Irony and Sarcasm Classification

Full Paper

Miljana Mladenović
Milenijum III
Vranje, Serbia
ml.miljana@gmail.com

Cvetana Krstev
University of Belgrade, Faculty of Philology
Belgrade, Serbia
cvetana@matf.bg.ac.rs

Jelena Mitrović
University of Passau, Faculty of Computer Science and Mathematics
Passau, Germany
jelena.mitrovic@uni-passau.de

Ranka Stanković
University of Belgrade, Faculty of Mining and Geology
Belgrade, Serbia
ranka@rgf.bg.ac.rs

ABSTRACT

The paper presents a language dependent model for classification of statements into ironic and non-ironic. The model uses various language resources: morphological dictionaries, sentiment lexicon, lexicon of markers and a WordNet based ontology. This approach uses various features: antonymous pairs obtained using the reasoning rules over the Serbian WordNet ontology (R), antonymous pairs in which one member has positive sentiment polarity (PPR), polarity of positive sentiment words (PSP), ordered sequence of sentiment tags (OSA), Part-of-Speech tags of words (POS) and irony markers (M). The evaluation was performed on two collections of tweets that had been manually annotated according to irony. These collections of tweets as well as the used language resources are in the Serbian language (or one of closely related languages – Bosnian/Croatian/Montenegrin). The best accuracy of the developed classifier was achieved for irony with a set of 5 features – (PPR, PSP, POS, OSA, M) – $acc = 86.1\%$, while for sarcasm the best results were achieved with the set (R, PSP, POS, OSA, M) – $acc = 72.8$.

CCS CONCEPTS

• **Computing methodologies → Artificial intelligence**; *Knowledge representation and reasoning*; *Semantic networks*; *Natural language processing*; *Lexical semantics*;

KEYWORDS

Computational irony, Verbal irony, Verbal Sarcasm, WordNet

ACM Reference format:
Miljana Mladenović, Cvetana Krstev, Jelena Mitrović, and Ranka Stanković. 2017. Using Lexical Resources for Irony and Sarcasm Classification. In *Proceedings of BCI '17, Skopje, Macedonia, September 20–23, 2017,* 8 pages.
https://doi.org/10.1145/3136273.3136298

BCI '17, September 20–23, 2017, Skopje, Macedonia

ACM ISBN 978-1-4503-5285-7/17/09…$15.00
https://doi.org/10.1145/3136273.3136298

1 INTRODUCTION

There are many different theories on what irony is and what role it plays in language understanding. According to [33] "Irony is . . . a uniquely human mode of communication, curious in that the speaker says something other than what he or she intends". Likewise, various classifications of irony as a rhetorical figure exist [27], but the most frequently used classification is the one differentiating between verbal, situational and dramatic irony. Verbal irony belongs to the group of rhetorical figures called Tropes [33] and it entails usage of a word in the way that is opposite of the presumed meaning.[1] In that regard we can say that ironic statement is the one where: (1) the receiver, apart from the sender, knows in advance which statement is true[2] (which sets irony apart from a lie), as well as that it is opposite of the expressed statement; (2) there are stylistic (usage of cursive font or quotation marks), syntactic (change of word order) or semantic (word polarity according to sentiment, rhetorical figure hyperbole, rhetorical question, antiphrasis) signals – markers [1], [5] which indicate the existence of irony.

Automatic irony detection is used more and more in NLP tasks, primarily for advancement of sentiment analysis systems, machine translation, authorship attribution, but also for systems analyzing linguistic structures at different levels – e.g. analysis of sentiment classification results in comments published on newspaper web portals, as it was done in [9] shows that 11% of comments from the set in question would be incorrectly marked as having positive polarity, if analysis and detection of the rhetorical figure irony had not been performed.

In this paper, we suggest and assess an automatic method of detection of verbal irony using rules of ontological reasoning in the Serbian WordNet (SWN) ontology. In Section 2 we present how irony detection problem was tackled in previous research. In many cases a corpus consisting of tweets was used, and we have developed a similar resource for Serbian which we present in Section 3. A system for recognition and tagging of ironic tweets based on the SWN ontology and other language resources is presented in Section 4. The results of the evaluation of the classifier, according to irony

[1] Ironic sentence "You are very smart!" with implied meaning e.g. "How did you manage to break this!?"
[2] Irony than leans on the implicit mechanism of presupposition (the implied information).

and sarcasm, are presented in Section 5, while some concluding remarks and plans for improvement are given in Section 6.

2 RELATED WORK

In automatic computational irony detection supervised machine learning and pattern matching techniques are used equally. Machine learning looks at a problem of verbal irony detection as a binary classification problem with classes of ironic and non-ironic statements. In the paper [27] comparison of Naive Bayes (NB) and Decision Trees (DT) methods used to classify tweets into ironic and non-ironic, showed that the results achieved by DT method measured by the $F1$ measure were better (71.5% and 53.3% for the balanced and imbalanced set of tweets, respectively). Findings reported in [3] depict binary classification of tweets using Random Forest (RF) and DT methods, where the best result ($F1 = 80\%$) was achieved using the RF method over the corpus of #humor tweets, while authors in [6] used Support Vector Machine, Logistic Regression (LR), DT, RF and NB, and the best result ($F1 = 74\%$) was achieved using the LR method with a combination of all suggested predictors. Pattern-based methods of irony detection utilize patterns either independent of or specific to a particular natural language that is being investigated. For example, authors in [31] used a corpus of tweets in Portuguese and patterns specific to the Portuguese language *so que*, *sim*, *na boa*, as well as language independent ones, like $(ADV^+ADV|ADJ^+ADJ)$[3] and $(!*|?*|!*?*|?*!*)$. Authors in [8] propose eight patterns out of which some are language dependent, and some are not, and point out that the best results were obtained with two independent patterns, marked as $P_{laugh} = (LOL|AH|EMO^+)$[4] and $P_{quote} = (ADJ|N)\{1,2\}$ leading to irony detection precision of 85.4% and 68.3%, respectively. In research described in [29] five linguistic patterns were suggested for recognition of ironic statements in a corpus of tweets in Chinese, while authors in [32] used the pattern "about as * as *" as a query sent to the Google Search engine, in order to obtain a corpus of examples of the rhetorical figure of comparison (simile) which was later used for classification into ironic and non-ironic comparisons. The results achieved by the classifier were $F1 = 73\%$ for detection of ironic examples and $F1 = 93\%$ for detection of non-ironic examples of comparisons.

3 CORPUS OF IRONY

One of the first challenges one encounters while trying to solve tasks of automatic recognition of verbal irony is selection of the collection of texts and marking ironic statements in it. For that purpose, online resources, such as Twitter, are used very frequently, where the hashtag #irony can be used to extract a tweet sub-collection which will be marked as ironic, while the non-ironic part can be formed out of tweets that contain other hashtags, such as #education, #politics, #health, etc. as described in [2], [19]. On the other hand, it is possible to mark a single Twitter account as a collection of ironic tweets, provided that the contents of that account are known [2], [15]. In a series of papers,[5] a collection of ironic tweets was formed using an extended set of hashtags – #irony, #sarcasm, #not, #yeahright – and tweets that contained the words that are normally attached to figurative usage "literally", "virtually", "figuratively" [14]. Authors in [29] used a microblogging platform called Plurk, similar to Twitter, to create a corpus of ironic statements. Other online resources can also be used for ironic corpus creation – Google Books search was used in the work by [21], where the phrase "said sarcastically" was used for generating a set of statements in which verbal irony appears. In [16], authors used the Google Search API to perform Google search, after which they used classification to form a corpus of ironic comparisons. A collection of comments on the popular German "news ticker" site was used in the work of [30], and comments from Portuguese online newspapers in the work of [8]. Finally, a crowdsourcing method was used to gather ironic statements from product ratings shown on the Amazon site [13].

For research presented in this paper, we have generated a corpus of tweets based on online search with geolocation and time constraints, using the query:

```
#ironija near:Belgrade,Serbia within:400km since:2013-01-01
        until:2015-10-29
```

We obtained 2,127 refined tweets after parsing. All links, hashtags and metadata were removed. We wanted to avoid a few problems in this process. The first one was related to the unification of language script, as the usage of Cyrillic and Latin scripts in Serbian is equal. All tweets were converted into Latin script.

The second problem was the classification of tweets according to language. Although our principal aim was to obtain a collection of tweets in Serbian, due to the fact that South Slavic languages Bosnian, Croatian, Serbian and Montenegrin (sometimes called BCMS)[6] are closely related, and most of the tweets in any of these languages can be considered understandable to a common speaker of Serbian, we did not use the `lang:sr` operator in the query itself because the corpus of tweets would be too small. However, geolocation restriction allowed us to also find tweets mostly written in the BCMS languages.

We developed a language tweet classifier that relies on lexical resources. Although resources we are using were developed for Serbian primarily, their development was based on traditional resources and texts covering to certain extent other related languages as well, making them suitable for this task. A language classifier was built and assessed in the following way (step 1 in Fig 1). First we manually marked each tweet with a (*BCMS*) or (*not_BCMS*) mark. After that we used Serbian Morphological Electronic Dictionaries [22] to automatically tag each word with a mark of belonging to a language `_word` or not belonging `_not` (resource A in Fig 1). We introduced a classification threshold as the smallest percentage of recognized words in a tweet. This classifier was applied for 8 different thresholds (Table 1) where positively classified tweets were marked with *cl* (recognized as tweets in BCMS), and negatively classified ones were marked with *ncl*. For the final set we chose the one that was obtained based on the recognition threshold of 40%, which means that each tweet from that set was classified positively if at least 40% of words in it were marked with a tag `_word`. The reason for accepting this threshold is a high degree of true positives

[3] In this and following examples ADJ stands for an adjective, ADV for an adverb and N for a noun.

[4] AH – onomatopoeic utterances (ah,eh, hi), EMO – a set of emoticons depicting positive feelings, LOL – acronym for Laughing out Loud

[5] SemEval 2015, Task 11

[6] See for example Alexander. R., Elias-Bursać, E.: Bosnian, Croatian, Serbian, a Textbook With Exercises and Basic Grammar. University of Wisconsin Press (2010)

(*tp*) with double the less false negative tweets (*fn*) compared to the 50% threshold.

The most common cause for false negative tweets was: the usage of letters without diacritics (*s* instead of *š*, *c* instead of *č* or *ć*, *z* instead of *ž* and *d* or *dj* instead of *đ*), usage of transcribed words instead of words in the BCMS languages, e.g. *hepi* for *happy*, and the repetition of some letters for emphasis, e.g. *saaad* instead of *sad* 'now'. The tweet *a saaad, sredjivanje sobee, jupiii, bas sam hepii* ('and noooow, cleaning up the rooom, yippee, I am so happyy') illustrates all three sources of problems (underlined words). Namely, these words cannot be recognized as words from BCMS as they were not found in dictionaries, and as a consequence the tweet was rejected.

An example of a false positive tweet is a tweet in Slovenian *Zakaj smo se pa borili, a za to, da bo odločitve partije zje*** neko Ustavno Sodišče? Pa kaj potem, če mu ni dokazano, kriv je* 'Why did we fight, so that the party decisions would be fu*** by some Constitutional Court? So what if there was no evidence, he is guilty', where underlined words belong to BCMS as well (not necessarily with the same meaning). In the example *Testovi za AM, A1 I A2 kategoriju su vrhunska stvar koja može da ti se desi u životu*, 'Tests for AM, A1 and A2 category are the best thing that can happen to you in your life' although some abbreviations were not identified, the tweet was identified as positive. Likewise, the example *nekas, rWtdien atvainosies un pasaule atkal kWWs rožaina, vai ne?*, not belonging to BCMS, but probably to the Litvanian language, was correctly recognized as negative.

Table 1: Language classification, based on the percentage of recognized words (threshold) in a tweet.

threshold	cl	ncl	precision	recall	F1	acc
30	1,942	185	0.895	0.998	0.944	0.903
40	1,892	235	0.915	0.994	0.953	0.920
50	1,849	278	0.930	0.987	0.958	0.929
60	1,730	397	0.960	0.953	0.957	0.929
70	1,596	531	0.978	0.896	0.935	0.898
80	1,343	784	0.991	0.764	0.863	0.801
90	898	1,229	0.996	0.513	0.677	0.599
100	630	1,497	0.997	0.360	0.529	0.475

After language classification was performed, the total number of tweets that we used for further analysis was the sum of true positives and false negatives ($tp + fn$), which amounted to a total of 1,903 tweets. In the last step, we have made corrections to the set of tweets obtained in this way. In tweets in Serbian (and other related languages) two types of mistakes appear systemically: (a) authors of tweets do not always use diacritic signs; (b) authors do not use the beginning capital letter where it should be used according to spelling rules, e.g. for personal names. In the hope of obtaining a more accurate collection of ironic tweets, we used a similar strategy that is utilized in most spell checkers, namely, we tried to correct only those words that were not found in the Serbian e-dictionaries (tagged with `_not`; step 2 in Fig 1). In the case of those words, if they contained one or more of the "critical" Latin script letters – *c, s, z* – or a combination of letters – *dj* – they were replaced with an appropriate letter of the Latin script, if that would lead to getting a word from the e-dictionaries.

This approach is illustrated by the following example (incorrectly written words are underlined):

Before correction: *Tim koji trenira Sasa Djordjevic ne moze da pogodi za tri poena! (The team trained by Sasa Djordjevic cannot score for three points!)*

After correction: *Tim koji trenira Sasa Đorđević ne može da pogodi za tri poena!*

It can be seen that in the corrected tweet one word was not corrected – Sasa instead of Saša – because sasa is a Serbian word for "sea anemone". A similar strategy was used for correction of words with missing initial capital letter. One example is:

Before correction: *partizan u jsl ligi remizira i gubi a u evropi dobija i prvi na tabeli (partizan draws the game and loses in the jsl league but wins in europe and tops the leader board)*

After correction: *partizan u jsl ligi remizira i gubi a u Evropi dobija i prvi na tabeli*

In this example, one mistake was corrected – capital letter in Europe – while in the other case it was not corrected, because although Partizan is the name of the sports club, it is also a common noun meaning "partisan".

Manual classification to ironic and non-ironic tweets using a set obtained in the above described way was performed by two linguistic experts (step 3, Fig 1). Keeping in mind the fact that we used query that searched Twitter with geolocation and time constraints and not by a language constraint, this set could also contain tweets that do not belong to the BCMS languages. Thus, annotators put each tweet into classes (`BCMS`, `not_BCMS`, `ironic`, `non_ironic`). Inter-annotator agreement was assessed using the Kripendorff α-test (*Kalpha*).[7] Research by [17] about values of the *Kalpha* coefficient showed that agreements whose values are $\alpha \geq 0.667$ can be considered reliable, and the agreements whose values are $\alpha \geq 0.8$ can be considered very reliable. Inter-annotator agreement between two annotators working on annotation of ironic tweets can be considered reliable in our case, as the value we obtained measured at $\alpha = 0.759$.

Verbal irony is a figure of speech used to convey statements that are opposite from those that are supposed to be conveyed, which is why, in this paper, we want to find pairs of antonymous concepts that can be used for detection and understanding of ironic constructs. There are not many direct antonyms in a natural language, therefore, their number is also small in the lexical-semantic network WordNet, compared to other relations. Also, indirect antonyms are often used in natural language, that is to say, synonyms of direct antonyms – e.g. in Princeton WordNet, adjectives *beautiful* and *ugly* are defined as direct antonyms which in the case of the example "a beautiful painting and an ugly painting" can be interchanged with a pair of indirect antonyms *beautiful* and *unpleasant* as in the example "a beautiful painting and an unpleasant painting", where *unpleasant* is a direct antonym of an adjective *pleasant* which is (in some context) a synonym of an adjective *beautiful*.

[7] Value of the $Kalpha$ coefficient can be in the interval [0, 1] where $Kalpha = 1$ represents complete agreement, and $Kalpha = 0$ represents complete disagreement. $Kalpha$ can also have a negative value from the $[-1, 0)$ interval caused by sampling mistakes or systemic disagreement.

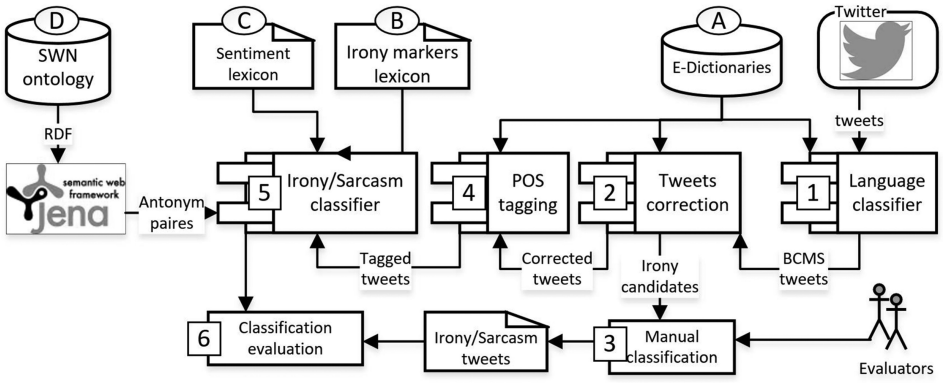

Figure 1: Architecture of the ironic/sarcastic tweets classifier.

In the Serbian WordNet ontology (resource D, Fig 1) there are currently 793 antonymous synset pairs defined with the *near_antonym* relation. In this paper we wish to define the rules that can be used for generating antonymous pairs in the SWN ontology that are not based solely on the relation of direct antonymy. By doing that, we would expand the set of indicators of ironic constructs. Relations that can participate in finding a broader set of synonyms and other word forms pertaining to a certain concept in the SWN ontology are: *synonym, similar_to, also_see, verb_group, be_in_state, hyponym.* Some of these relations connect the same POS synsets, the others are cross-POS (like *be_in_state*). Lexical relations such as *derived, derived-vn, derived-gender, derived-pos, attribute and particle* [20] can also be used, but since they are not frequent in SWN we will use in this paper only the first six.[8] Figure 2 shows an example of synsets and relations between them in the SWN ontology which are used for defining two separate sets of mutually indirect antonymous concepts.

4 IRONY CLASSIFIER

Reasoning rules in the SWN ontology related to the existence of ironic pairs in which the six previously mentioned relations participate and which can be used for generating "synonyms" of one member of the relation of direct antonymy are depicted here in the form of Jena rules:[9]

```
"[rule1: (?a swn:synonym ?b) (?b near_antonym: ?c)
-> (?a swn:irony ?c)]"

"[rule2: (?a swn:similar_to ?b) (?b near_antonym: ?c)
-> (?a swn:irony ?c)]"

"[rule3: (?a swn:also_see ?b) (?b near_antonym: ?c)
-> (?a swn:irony ?c)]"

"[rule4: (?a swn:verb_group ?b) (?b near_antonym: ?c)
-> (?a swn:irony ?c)]"

"[rule5: (?a swn:bee_in_state ?b) (?b near_antonym: ?c)
-> (?a swn:irony ?c)]"

"[rule6: (?a swn:hyponym ?b) (?b near_antonym: ?c)
-> (?a swn:irony ?c)]"
```

If we look at the rules from the set $\{rule2, \ldots, rule6\}$, it can be noticed that each rule can be expanded with the relation *synonym*, if it exists, so that we get an expanded set of rules in the following form:

```
"[rule21: (?a swn:synonym ?b)(?b swn:similar_to ?c)
(?c near_antonym:?d)->(?a swn:irony ?d)]"

"[rule31: (?a swn:synonym ?b)(?b swn:also_see ?c)
(?c near_antonym:?d)->(?a swn:irony ?d)]"

"[rule41: (?a swn:synonym ?b)(?b swn:verb_group ?c)
(?c near_antonym:?d)->(?a swn:irony ?d)]"

"[rule51: (?a swn:synonym ?b)(?b swn:bee_in_state ?c)
(?c near_antonym:?d)->(?a swn:irony ?d)]"

"[rule61: (?a swn:synonym ?b)(?b swn:hyponym ?c)
(?c near_antonym:?d)->(?a swn:irony ?d)]"
```

In the same way the rules can be expanded with a mutual combination of all given relations, where the number of relations in a rule goes up to a maximum of all six relations. Restrictions that we introduce are the following: (1) a relation cannot be repeated more than once in a given rule; (2) the relation *synonym* can be found only as the first one in a series of given relations. That is how we have obtained 471 reasoning rules for retrieving an indirect member of the antonymous relation, that is to say, for establishing an ironic relationship with the opposite member of the antonymous relation. A part of the set of rules that begin with the second relation *similar_to* is given below.

```
"[rule2:][rule21:][rule23:][rule24:][rule25:][rule26:]
[rule231:]...[rule265341:][rule265431:]"
```

We have applied the set of 471 rules over each one of the 793 antonymous pairs of synsets defined using the *near_antonym* relation, thus obtaining a set of 3,258 pairs (a, z) of antonymous concepts acquired

[8] Total number of synsets per relation in SWN: synonym 14,239, similar_to 222, also_see 215, verb_group 185, be_in_state 288, hyponym 20,709.

[9] We used the following software tools in this paper: Developing tool Eclipse Java EE IDE Luna and Apache Jena open source software development environment which allows for reasoning at the level of OWL 2 language by converting OWL rules into the Jena rules format.

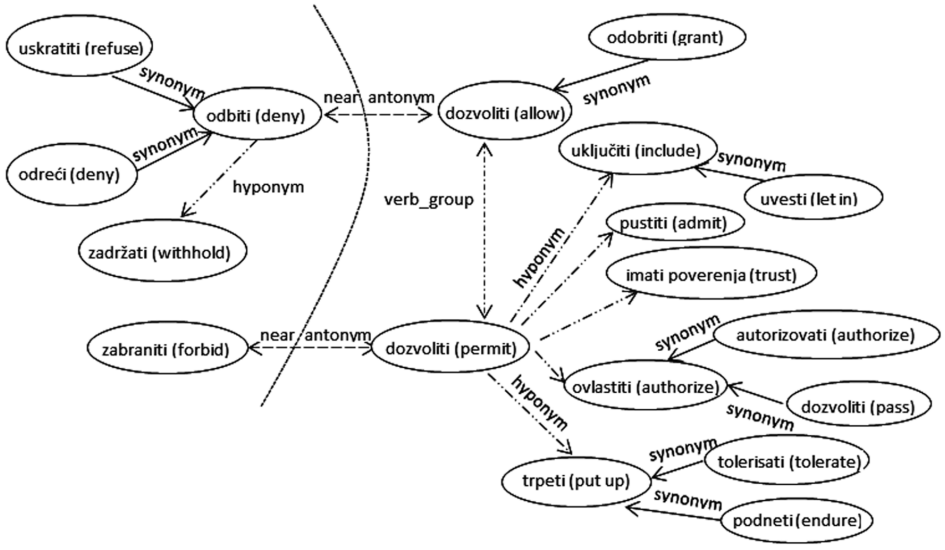

Figure 2: Verb synsets in the SWN ontology mutually connected with relations that participate in finding a broader set of synonyms and separated by the antonymy relation.

from RDF triples (`?a swn:irony ?z`), where a is the first and z is the last member of the observed rule.

According to [1], existence of irony in a text is characterized by markers. Those are literary devices which indicate that irony is present, but if we remove them, ironic meaning does not change. In the work described by [23] irony markers in BCMS, such as the usage of particles *baš* and *već*[10] and stylistic irony markers like punctuation marks, exclamation mark, and cursive font were described. Still, irony markers can also be phrases such as: [*Uh*|*Ah*] *što volim* '[Uh|Ah] I really love that', *Ah, kakav...* 'Ah, what a ...', *nema ničeg lepšeg* 'there is nothing more beautiful',[*Ma*|*Pa*|*Baš*] *bravo* '[And|So|Just] bravo', *bolje ne može* 'it could not be better', *ultra-mega-giga* 'ultra-mega-giga', etc. A complete set of irony markers in lexicon form (resource B in Fig. 1) is a part of the architecture of the suggested model.

Ironic tweet classifier (Fig 1) for the purpose of feature construction uses: (1) a set of antonymous pairs (a, z) obtained from the SWN ontology (resource D) a lexicon of irony markers (resource B) a sentiment lexicon (resource C), POS tagger markers (obtained in step 4 from resource A). Due to the fact that SWN ontology contains only noun, verb, adjective and adverb synsets, as those POS are represented in the SWN, the antonymous pairs (a, z) are limited to these word forms. In that regard, we needed a POS tagger to analyze each tweet only at the level of this set of word forms. For that purpose, we used a hybrid system for Serbian that combines three NLP tasks: PoS tagging, compound and named-entity recognition [10] (step 5 in Fig. 1) that was trained on various annotated texts – literary, newspaper and textbooks. Tagging results are represented by two previously given sentences (double-underlined are incorrectly tagged words, single-underlined are incorrectly classified as common words):

Tim_PRO koji_PRO trenira_V –_PONCT
Saša_DJordjević_NEpers ne_PAR može_V da_CONJ pogodi_V
za_PREP –_PONCT tri_poena_NEamount [11]

partizan_N u_PREP jsl_PRO ligi_N remizira_V i_CONJ gubi_V
a_CONJ u_PREP –_PONCT Evropi_NEtop dobija_V i_CONJ
prvi_A na_PREP tabeli_N [12]

5 EVALUATION

5.1 The classifier of irony

Annotation of each tweet was twofold: the annotators were asked to decide whether the language of the tweet was recognized and whether the tweet represents an ironic statement.[13] The results of the language tagging were used to estimate a binary language classifier (*BCMS* or *not_BCMS*). After the language classification we obtained a subset of 1,732 tweets that were automatically annotated as tweets in *BCMS*. That set of tweets was then used in an evaluation of the irony classifier. In it, 319 tweets were tagged as ironic by both annotators and we treated them as ironic in a further process. Tweets tagged by only one annotator as ironic were treated as not ironic. In that way we obtained the imbalanced set to evaluate the performance of our automatic classifier and its ability to classify into ironic and non-ironic tweets (step 6 in Fig 1).

Prior to classification a set of indirect antonymous pairs was generated (Section 3), e.g. the adjective *ilegalan* 'illegal' was related not only to its direct antonyms *zakonski, legalan* 'legal' but also to indirect anonymous adjectives *zakonit* 'lawful', *regularan, redovan* 'regular' and a noun *zakonitost* 'lawfulness'. This resource allows

[10] E.g. *Baš si dobar prijatelj!* 'You are indeed a really good friend!' and *Već si sve uredio!* 'You have already taken care of everything!'

[11] Eng. The team trained by Saša Djordjević cannot score for three points.

[12] Eng. partizan draws the game and loses in the jsl league but wins in Europe and tops the leader board.

[13] Annotated data are available at http://ankete.mmiljana.com under the terms of the CC_BY-NC licence.

detection of words that have a capacity to participate in the formation of ironic statements, due to the fact that we know that they have a corresponding antonym. The other resource used to detect the occurrence of irony is a lexicon of sentiment words and phrases in Serbian (resource C, Fig. 1). Keeping in mind the nature of the rhetorical figure verbal irony which is used to portray a negative statement in the form of a positive one, using the sentiment lexicon we can detect words and phrases that carry positive sentiment polarity. We have used in this research the sentiment lexicon developed for sentiment analysis and described in [24]. The lexicon contains 4,593 entries with sentiment polarity values. Lexicon of irony markers (resource B, Fig. 1) which consists of 62 phrases, whose examples we quoted in the previous section, was built based on research presented in [18], [25], [26]. Finally, we used the results of the POS tagger so that we could experiment with different POS in the process of generating classifier features. Basic forms of words (lemmas) were used in all cases.

The following features were used in the classification process: antonymous pairs (R), antonymous pairs where one member has positive sentiment polarity (PPR), polarity of positive sentiment words (PSP), POS tags of words (POS), ordered sequence of sentiment tags (P – positive, N – negative, z – unknown or neutral) created based on sentiment polarity of the sequence of words in a tweet (OSA) and irony markers (described in Section 3) (M). We used these features to train a MaxEnt classifier, a supervised machine learning algorithm which we implemented using MaxEnt SharpEntropy library[14] on 5-folded cross-validated dataset. The classification results according to the set of applied features are given in Table 2.

Table 2: Results of Irony classification on Twitter data.

	feature set	P	R	F1	acc
FS1	POS, OSA, M	0.504	0.530	0.517	0.817
FS2	R, OSA, M	0.605	0.486	0.539	0.845
FS3	PSP, POS, OSA, M	0.616	0.473	0.535	0.849
FS4	R, PSP, POS, OSA, M	0.658	0.458	0.540	0.856
FS5	PPR, POS, OSA, M	0.670	0.458	0.544	0.858
FS6	PPR, PSP, POS, OSA, M	0.686	0.451	0.544	0.861

The best recall was achieved with the feature set (POS, OSA, M) in which positive sentiment words were not used as a feature, nor were there any antonymous pairs, as can be seen from Table 2. Still, due to lower precision, accuracy of this classifier ($acc = 0.817$) was also lower than in the next five experiments. In the second experiment, which used the set of features containing antonymous pairs, ordered sequence of sentiment tags in a tweet and irony markers (R, OSA, M) increased precision, but lowered recall. In the third experiment, adding the polarity of positive sentiment words (PSP) improved the results, which was also the case with the fourth experiment, where antonymous pairs (R) were added as a feature and the classification accuracy was better as well. In the last two experiments, when the set of antonymous pairs was substituted by the set in which one member had positive sentiment polarity (PPR), we obtained better accuracy, while the last experiment using the set of five features (PPR, PSP, POS, OSA, M) gave the best results of this classifier ($acc = 86.1\%$), for values $tp = 144$, $fp = 66$, $fn = 175$, $tn = 1,347$.

Downsides of this type of classification, in a general case, lie in the limited nature of the resources (sentiment lexicon, set of rules used in generating antonymous pairs, WordNet ontology) that are being used. Still, with the help of antonymous pairs, we can comprehend the real meaning of an ironic statement, like in the following example:

> *Bila sam ljubomorna, a onda sam je videla.. od tad više nisam ljubomorna.. toliko si lepa da te ni promaja ne bi udarila.* (lep – ružan) 'I was jealous and then I saw her..since then I am no longer jealous..you are so beautiful that even the draught wouldn't hit you' (*beautiful – ugly*)

where it can be determined that in the given tweet there is an adjective *lepa* 'beautiful' which is present in two antonymous pairs: same-POS (*lep – ružan* 'beautiful – ugly') and cross-POS (*lep – ružnoća* 'beautiful – ugliness'), which is why the other member of the pair can be found automatically (first between the same POS pairs, if there are any), which, in this case, is the word *ružan* 'ugly'. The tweet can therefore be interpreted with an opposite – and intended – meaning: *toliko si ružan da te ni promaja ne bi udarila* 'you are so ugly that even the draught wouldn't hit you'.

5.2 The application of the irony classifier to sarcasm

A thin line divides ironic utterances from sarcastic ones. There are many definitions and comparisons of these two figures. According to one of the most comprehensive studies, given by the Inkpot group [11] within the Rhetfig project,[15] which combines linguistic and rhetorical theories with discourse analysis and machine learning to develop formal models of computational rhetoric, one definition of sarcasm is "Use of mockery, verbal taunts, or bitter irony". For this reason, we explored whether the irony classification system can be used for sarcasm classification. The other reason for this experiment are promising results presented in [7],[34],[4],[12],[28], where irony and sarcasm were treated in the same way in classification tasks.

Similarly as in the case of building a corpus of ironic tweets, a corpus of sarcastic tweets has been generated based on online search with geolocation and time constraints, using the hashtag #sarkazam(sarcasm)

```
#sarkazam near:Belgrade,Serbia within:400km since:2013-01-01
          until:2015-10-29
```

The rest of the treatment was the same as in the case of ironic tweets and it was done following the next steps:

(1) all tweets written in Cyrillic have been automatically converted into Latin script;
(2) all tweets in the corpus of sarcastic tweets have been automatically classified into two classes: BCMS language and

[14] https://sharpnlp.codeplex.com

[15] http://rhetfig.appspot.com

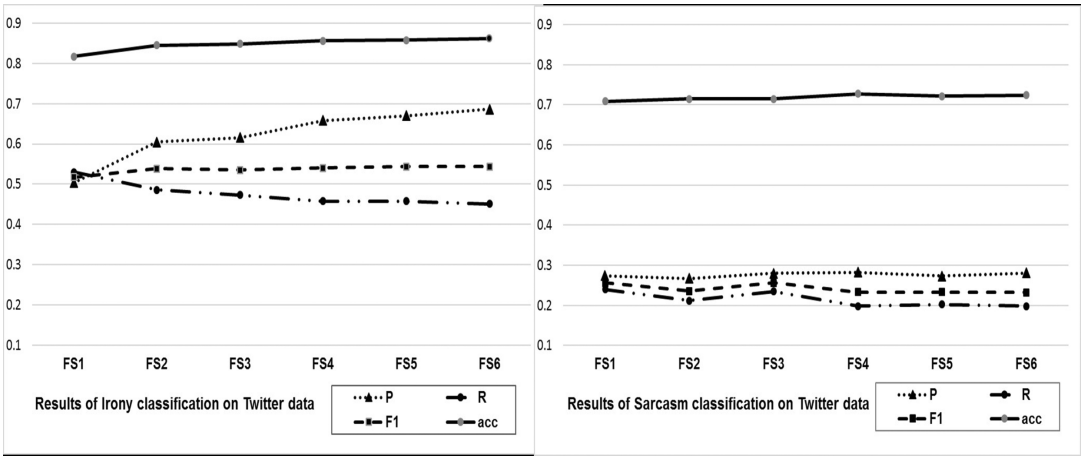

Figure 3: Performance measures of the ironic/sarcastic tweets classifier depending on feature sets.

not_BCMS language; after the language-dependent classification was performed, the total number of tweets that we used for further analysis was the sum of true positives and false negatives ($tp + fn$), which amounted to a total of 1,167 tweets;

(3) manual classification to sarcastic and non-sarcastic tweets was performed by the same linguistic experts and inter-annotator agreement, assessed using the Kripendorff α-test, achieved $\alpha = 0.86$; we used, for further processing, those tweets which have been assessed by both annotators as either sarcastic or non-sarcastic; in that way we obtained a total of 1,042 tweets, where 825 were assessed by both annotators as non-sarcastic and 217 as sarcastic;

(4) Like in the case of irony, we used results of the POS tagger for experimenting with different POS in the process of generating classifier features;

(5) We used that imbalanced set of 1,042 tweets to evaluate the performance of our automatic classifier and its ability to classify into sarcastic and non-sarcastic tweets.

The features used for classification of sarcastic tweets were the same as in the case of classification into ironic and non-ironic tweets: antonymous pairs (R), antonymous pairs where one member has positive sentiment polarity (PPR), polarity of positive sentiment words (PSP), PoS tags of words (PoS), ordered sequence of sentiment tags (P-positive, N-negative, z-unknown or neutral) created based on sentiment polarity of the sequence of words in a tweet (OSA) and irony markers (M). Classification results according to the set of applied features are given in Table 3.

Although irony and sarcasm have similar functions in a natural language, sarcasm classification results given in Table 3 show that language resources used for creating features have to be changed and adjusted. To this end, a modified list of stylistic and semantic markers for a sarcasm detection has to be improved and the sentiment lexicon has to be changed to contain hateful and offensive words and phrases. But, like in the case of irony classification, when a tweet is interpreted as sarcastic, the classifier offers, with the help of antonymous pairs, the real meaning of a sarcastic tweet, like in the following example:

Table 3: Results of Sarcasm classification on Twitter data.

	feature set	P	R	F1	acc
FS1	POS, OSA, M	0.274	0.240	0.256	0.709
FS2	R, OSA, M	0.267	0.212	0.236	0.715
FS3	PSP, POS, OSA, M	0.280	0.235	0.256	0.715
FS4	R, PSP, POS, OSA, M	0.283	0.198	0.233	0.728
FS5	PPR, POS, OSA, M	0.273	0.203	0.233	0.722
FS6	PPR, PSP, POS, OSA, M	0.281	0.198	0.232	0.724

> *Sreća pa zimus neće biti uglja, gasa, struje i ogreva jer sa ovim smanjenjem ne bi ni moglo da se plati!* (sreća - žalost)
> 'Fortunately this winter there will be no coal, gas, electricity or fuel, because with this pay cut it would be impossible to pay!' (*Fortunately – Unfortunately*)

By selecting a proper set of features, the precision of the ironic tweets classifier can be notably improved (Fig. 3), but in the case of sarcasm, all sets of features provide similar and low performance measures. The results of both classifiers show low grade of recall and it shows us that we should enlarge and improve lexical resources that have been used and expand the set of ontological rules to include procedures of generalization and specifications of concepts that are already included by the rules laid down by this system. Also, we should use more balanced datasets to achieve more reliable results.

6 CONCLUSIONS

In this paper we have suggested and assessed language dependent classification and correction of the corpus of tweets intended to be used in the process of classifying tweets into ironic and non-ironic ones. We have offered a model of irony classification, using the following features: antonymous pairs obtained using the reasoning

rules over the Serbian WordNet ontology, the same antonymous pairs in which one member has positive sentiment polarity, positive sentiment polarity of words, ordered sequence of sentiment tags in a tweet, POS tags of words and irony markers. We have shown that integration of a feature represented by antonymous pairs where one member has positive sentiment polarity can improve classification compared to the case when that feature is not used. The performance of the classifier was assessed for different sets of features, and the best result (precision=68.6%, acc=86.1%) was achieved with the set of 5 features. Taking into account the fact that we used an imbalanced dataset (319 ironic and 1,413 non-ironic tweets), comparison with other similar irony classifiers [27] confirms that when using an imbalanced distribution, the accuracy is higher relative to precision, recall and $F1$. Our results also show that semantic knowledge in the WordNet ontology can improve irony classification. In the process of obtaining the set of antonymous pairs we have used six relations for finding a broader set of synonyms and other word forms related to a certain concept in the SWN ontology. The results of the classification according to sarcasm by using the developed irony classifier were not satisfactory and show that language resources have to be improved.

In future work, we plan on broadening the set of antonymous pairs using ontological rules in which other lexical relations participate. We will investigate other sets of features and generate other textual collections that can be used for classification according to irony. Also, experiments without previous correction of spelling errors, to better predict the real life performance of the method, will be performed.

ACKNOWLEDGMENTS

We are grateful to Marija Pantić, Milica Andrić and Ana Barbatesković, master students at the University of Belgrade for manual classification of tweets. This research was partially supported by Serbian Ministry of Education and Science under the grants #III 47003 and 178003.

REFERENCES

[1] Salvatore Attardo. 2000. Irony markers and functions: Towards a goal-oriented theory of irony and its processing. *Rask International Tidsskrift for Sprog og Kommunication* 12, 1 (2000), 3–20.

[2] Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. 2014. Italian Irony Detection in Twitter: a First Approach. In *The First Italian Conference on Computational Linguistics CLiC-it 2014 & the Fourth International Workshop EVALITA*. 28–32.

[3] Francesco Barbieri and Horacio Saggion. 2014. Modelling Irony in Twitter: Feature Analysis and Evaluation. In *9th LREC*. 4258–4264.

[4] Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014. Modelling sarcasm in twitter, a novel approach. In *Association for Computational Linguistics*. 50–58.

[5] Christian Burgers, Margot van Mulken, and Peter Jan Schellens. 2013. The use of co-textual irony markers in written discourse. *Humor* 26, 1 (2013), 45–68.

[6] Konstantin Buschmeier, Philipp Cimiano, and Roman Klinger. 2014. An impact analysis of features in a classification approach to irony detection in product reviews. In *Proc. of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA) – 52th ACL*. 42–49.

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

[8] Paula Carvalho, Luís Sarmento, Mário J Silva, and Eugénio De Oliveira. 2009. Clues for Detecting Irony in User-generated Contents: Oh...!! It's so easy;-). In *Proc. of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*. ACM, 53–56.

[9] Paula Carvalho, Luís Sarmento, Jorge Teixeira, and Mário J Silva. 2011. Liars and Saviors in a Sentiment Annotated Corpus of Comments to Political Debates. In *Proceedings of the 49th Annual Meeting of the ACL: Human Language Technologies: short papers – Volume 2*. Association for Computational Linguistics, 564–568.

[10] Matthieu Constant, Cvetana Krstev, and Duško Vitas. 2015. Hybrid Lexical Tagging in Serbian. In *Proc. of 7th Language & Technology Conference*. Fundacja Uniwersytetu im. A. Mickiewicza, Poznań, 461–465.

[11] Chrysanne DiMarco and Randy Allen Harris. 2011. The RhetFig Project: Computational Rhetorics and Models of Persuasion. *Informatica* (2011).

[12] Elisabetta Fersini, Federico Alberto Pozzi, and Enza Messina. 2015. Detecting irony and sarcasm in microblogs: The role of expressive signals and ensemble classifiers. In *Data Science and Advanced Analytics (DSAA), 2015. 36678 2015. IEEE International Conference on*. IEEE, 1–8.

[13] Elena Filatova. 2012. Irony and Sarcasm: Corpus Generation and Analysis Using Crowdsourcing. In *8th LREC*. 392–398.

[14] Aniruddha Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. 2015. Semeval-2015 task 11: Sentiment analysis of figurative language in Twitter. In *Proc. of the 9th Int. Workshop on Semantic Evaluation (SemEval 2015)*. 470–478.

[15] Andrea Gianti, Cristina Bosco, Viviana Patti, Andrea Bolioli, and Luigi Di Caro. 2012. Annotating irony in a novel italian corpus for sentiment analysis. In *Proc. of the 4th Workshop on Corpora for Research on Emotion Sentiment and Social Signals*. 1–7.

[16] Yanfen Hao and Tony Veale. 2010. An Ironic Fist in a Velvet Glove: Creative Mis-representation in the Construction of Ironic Similes. *Minds and Machines* 20, 4 (2010), 635–650.

[17] Andrew F Hayes and Klaus Krippendorff. 2007. Answering the Call for a Standard Reliability Measure for Coding Data. *Communication methods and measures* 1, 1 (2007), 77–89.

[18] Virna Karlić and Goran Koletić. 2013. Explicit Verbal Irony and the Means of Marking It in the Journalistic Style of the Serbian and Croatian Languages. *Zeitschrift für Balkanologie* 49, 2 (2013).

[19] Jihen Karoui, Farah Benamara Zitoune, Véronique Moriceau, Nathalie Aussenac-Gilles, and Lamia Hadrich Belguith. 2015. Towards a Contextual Pragmatic Model to Detect Irony in Tweets, In 53rd Annual Meeting of the ACL and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of NLP (ACL 2015). *Volume 2: Short Papers*, 644–650.

[20] Svetla Koeva, Cvetana Krstev, and Duško Vitas. 2008. Morpho-semantic relations in Wordnet – a case study for two Slavic languages. In *Global Wordnet Conference (GWC'08)*. University of Szeged, Department of Informatics, 239–253.

[21] Roger J. Kreuz and Gina M. Caucci. 2007. Lexical Influences on the Perception of Sarcasm. In *Proc. of the Workshop on Computational Approaches to Figurative Language (FigLanguages '07)*. Association for Computational Linguistics, 1–4.

[22] Cvetana Krstev. 2008. *Processing of Serbian — Automata, Texts and Electronic Dictionaries*. Faculty of Philology, University of Belgrade, Belgrade.

[23] Mirjana Mišković. 2001. The particle *baš* in contemporary Serbian. *Pragmatics* 11, 1 (2001), 17–30.

[24] Miljana Mladenović, Jelena Mitrović, Cvetana Krstev, and Duško Vitas. 2015. Hybrid sentiment analysis framework for a morphologically rich language. *Journal of Intelligent Information Systems* 46, 3 (2015), 599–620.

[25] Silvana Neshkovska. 2015. Stylistic Signals of Verbal Irony. *International Journal of Language & Linguistics* 2, 2 (2015).

[26] Nikolina Palašić. 2015. Komunikacijska vrijednost ironije. *Fluminensia: Časopis za filološka istraživanja* 27, 1 (2015), 123–136.

[27] Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in Twitter. *Language Resources and Evaluation* 47, 1 (2013), 239–268.

[28] Emilio Sulis, Delia Irazú Hernández Farías, Paolo Rosso, Viviana Patti, and Giancarlo Ruffo. 2016. Figurative messages and affect in Twitter: Differences between #irony, #sarcasm and #not. *Knowledge-Based Systems* 108 (2016), 132–143.

[29] Yi-jie Tang and Hsin-Hsi Chen. 2014. Chinese Irony Corpus Construction and Ironic Structure Analysis. In *The 25th International Conference on Computational Linguistics (COLING)*. 1269–1278.

[30] Bianka Trevisan, Melanie Neunerdt, Tim Hemig, Eva-Maria Jakobs, and Rudolf Mathar. 2014. Detecting Irony Patterns in Multi-level Annotated Web Comments. In *Workshop Proceedings of the 12th KONVENS 2014*. 34–41.

[31] Aline A Vanin, Larissa A Freitas, Renata Vieira, and Marco Bochernitsan. 2013. Some Clues on Irony Detection in Tweets. In *Proc. of the 22nd International Conference on World Wide Web*. 635–636.

[32] Tony Veale and Yanfen Hao. 2010. Detecting Ironic Intent in Creative Comparisons. In *19th European Conference on Artificial Intelligence (ECAI)*, Vol. 215. 765–770.

[33] Byron C Wallace. 2015. Computational irony: A survey and new perspectives. *Artificial Intelligence Review* 43, 4 (2015), 467–483.

[34] Po-Ya Angela Wang. 2013. #Irony or #Sarcasm – A Quantitative and Qualitative Study Based on Twitter. In *Proceedings of the PACLIC: the 27th Pacific Asia Conference on Language, Information, and Computation*. Department of English, National Chengchi University, 349–356.