10.12871/clicit2014211

ITGETARUNS A Linguistic Rule-Based System for Pragmatic Text Processing

Rodolfo Delmonte
Dipartimento di Studi Linguistici e Culturali Comparati
Ca' Bembo – Dorsoduro 1075
Università Ca' Foscari – 30123 VENEZIA, Italy
delmont@unive.it

Abstract

English. We present results obtained by our system ITGetaruns for all tasks. It is a linguistic rule-based system in its bottom-up version that computes a complete parser of the input text. On top of that it produces semantics at different levels which is then used by the algorithm for sentiment and polarity detection. Our results are not remarkable apart from the ones related to Irony detection, where we ranked fourth over eight participants. The results were characterized by our intention to favour Recall over Precision and this is also testified by Recall values for Polarity which in one case rank highest of all.

Italiano. *Presentiamo i risultati ottenuti dal nostro sistema ITGetaruns per tutti i task. Si tratta di un sistema basato su regole linguistiche nella sua versione bottom-up, che produce un parse complete del testo in ingresso. Al di sopra di questo produce semantica a diversi livelli, che viene poi usata dall'algoritmo per l'analisi della polarità e della soggettività. I nostri risultati non sono notevoli a parte quelli relativi alla individuazione dell'Ironia, nella quale ci siamo classificati quarti su sette partecipanti. I risultati sono caratterizzati dalla nostra intenzione di favorire il Recall sulla Precision and questo è anche documentato dai valori della Recall per la polarità che in un caso sono i più alti in assoluto.*

1 Description of the System

The system we called ITGetaruns shares its backbone with the companion English system which has been used – and documented – for a number of international challenges on Semantic and Pragmatic computing in English texts. It is organized around a manually checked subcategorized lexicon, a sequence of rules organized according to theoretical linguistics criteria and combines data-driven (bottom-up) and grammar-driven (top-down) techniques.

Technically speaking, it is based on a shallow parser, which in turn is based on a chunker and NER and multiword recognizer. On top of this parser, there is constituent or phrase structure parser, which sketches sentence structure. This is then passed to a deep dependency parser, which combines constituent level information, lexical information, and a Deep Island Parser. The aim of this third parser is that of producing semantically viable Predicate-Argument Structures. Finally, on top of this level of representation, the Pragmatic System is built.

Conceptually speaking, the deep island parser (hence DIP) is very simple to define, but hard to implement. A semantic island is made up by a set of A/As, which are dependent on a verb complex (hence VCX). Arguments and Adjuncts may occur in any order and in any position: before or after the verb complex, or be simply empty or null. Their existence is determined by constituents surrounding the VCX. The VCX itself can be composed of all main and minor constituents occurring with the verb and contributing to characterize its semantics. We are here referring to: proclitcs, negation and other adverbials, modals, restructuring verbs (*lasciare*/let, *fare*/make, etc.), and all auxiliaries. Tensed morphology can then appear on the main lexical verb or on the auxiliary/ modal/ restructuring verb. Gender can appear on the past participle when the verb takes auxiliary ESSERE, or when a complement is duplicated by Clitic Left Dislocation.

The DIP is preceded by a tagger, which is accompanied by a multiword expression labeller. Tagged input is passed to an augmented context-free parser that works on top of a chunker. The chunker collects main constituents on the basis of a Recursive Transition Network of Italian and then passes the output to a cascaded sentence level parser. Constituents are labelled with usual grammatical relations on the basis of syntactic subcategorization contained in our verb lexicon of Italian counting some 17,000 entries. There are some 270 different syntactic classes, which differentiates also the most common prepositions associated to oblique arguments. Linear position and precedence in the input string is assumed at first as a valid criterion for distinguishing SUBJects from OBJects. Adjustments will be executed by the semantic parser, which will be responsible for the final relabeling of the output.

The DIP receives the output of the surface parser, a list of Referring Expressions and a list of VCX. Referring expressions are all nominal heads accompanied by semantic class information collected in a previous recursive run through the list of the now lemmatized and morphologically analysed input sentence. It also receives the output of the context-free parser. The DIP searches for SUBJects at first and assumes it is positioned before the verb and close to it. In case there is none such chunk available the search is widened if intermediate chunks are detected: they can be Prepositional Phrases, Adverbials or simply Parentheticals. If this search fails, the DIP looks for OBJects close after the verb then and again possibly separated by some intermediate chunk. They will be relabelled as Subjects. Conditions on the A/As boundaries are formulated in these terms: between current VCX and prospective argument there cannot be any other VCX. Additional constraints regard presence of relative or complement clauses, which are detected from the output chunked structure.

The prospective argument is deleted from the list of Referring Expressions and the same happens with the VCX. The same applies for the OBJect, OBJect1 and OBLique. When arguments are completed, the parser searches recursively for ADJuncts, which are PPs, using the same boundary constraint formulation above.

Special provisions are given to copulative constructions, which can often be reversed in Italian: the predicate coming first and then the subject NP. The choice is governed by looking at referring attributes, which include definiteness, quantification, distinction between proper/common noun. It assigns the most referring nominal to the SUBJect and the less referring nominal to the predicate. In this phase, whenever a SUBJect is not found from available referring expressions, it is created as little_pro and morphological features are added from the ones belonging to the verb complex. After updating of the Referring Expressions with the new Grammatical Relations, the parser searches the most adequate Semantic Role to be associated to it. This is again taken from a lexicon of corresponding verb predicates and works according to the type of overall Predicate-Argument Structure (hence PAS).

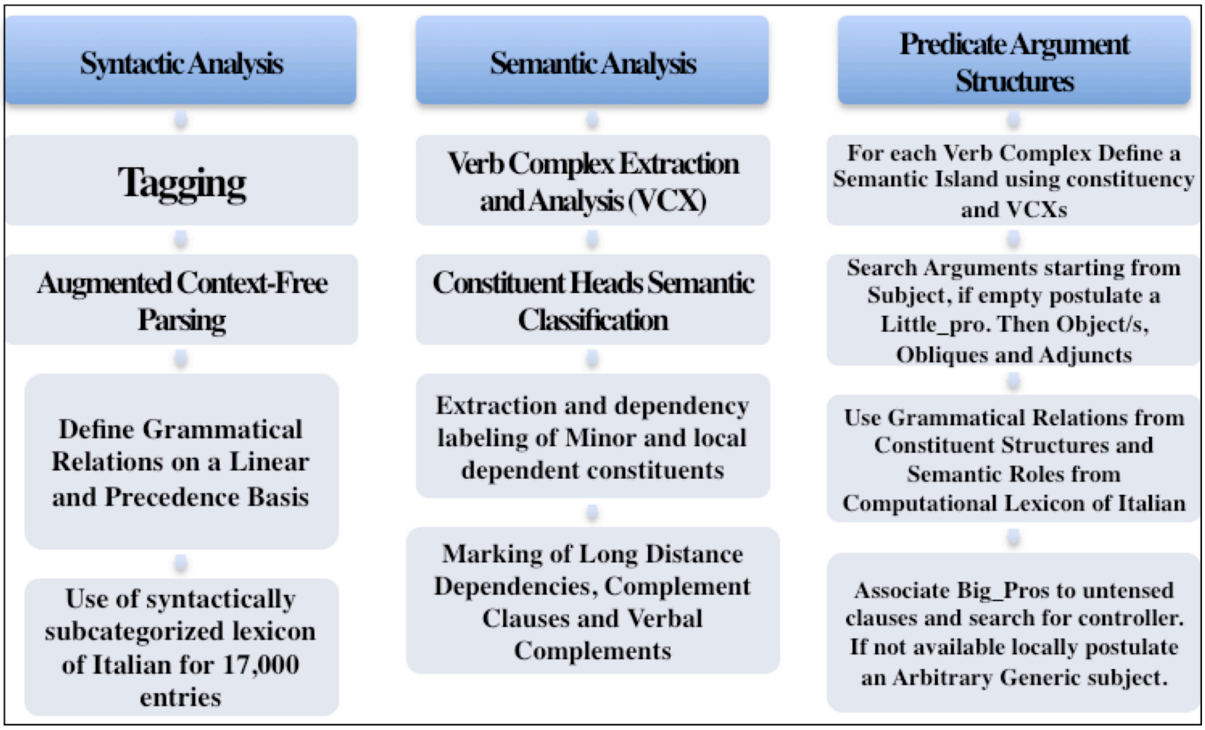

Table 1. Flowchart of modules for Deep Island Parser.

The SUBJect is in fact strictly depending on the semantics associated to the verb, but in case of ambiguity the system delays the assignment of semantic role until a complete PAS is obtained. In this phase, passive diathesis is checked in order to apply a lexical rule from LFG, that assigns OBJect semantic role to the SUBJect of the corresponding passive form of the verb predicate.

The PAS thus obtained, is then enriched by a second part of the algorithm, which adds empty or null elements to untensed clauses. The system starts from little_pros and looks for local possible antecedents. An additional semantic function is activated in this phase of analysis and is the creation of verbal multiwords, constituted by the concatenation of a verb lemma and the head of its object, as for instance "tener conto"/take_into_account, which transforms the main predicate TENER into TENER_CONTO. In this operation, the system has available a list of light verbs of Italian which are the most frequent main component of the compound: then the OBJect complement head is extracted and the concatenation is searched in a specialized dictionary of verbal multiwords of Italian. The OBJect is then erased from the list of arguments and the Argument/Adjunct distinction is updated according to the new governing predicate.

1.1 The Pragmatic Parser

Measuring the polarity of a text is usually done by text categorization methods which rely on freely available resources. However, we assume that in order to properly capture opinion and sentiment (see Delmonte & Pallotta 2011; Kim & Hovy 2004; Pang & Lee 2004; Wiebe et al. 2005), expressed in a text or dialog, - that we also assume to denote the same field of research, and is strictly related to "subjectivity" analysis - any system needs a linguistic text processing approach that aims at producing semantically viable representation at propositional level. In particular, the idea that the task may be solved by the use of Information Retrieval tools like Bag of Words Approaches (BOWs) is insufficient. BOWs approaces are sometimes also camouflaged by a keyword based Ontology matching and Concept search (see Kim and Hovy 2004), based on SentiWordNet (see Esuli & Sebastiani 2006) more on this resource below -, by simply stemming a text and using content words to match its entries and produce some result (Turney and Littman 2003). Any search based on keywords and BOWs is fatally flawed by the impossibility to cope with such fundamental issues as the following ones, which Polanyi & Zaenen (2006) named contextual valence shifters:

- presence of negation at different levels of syntactic constituency;
- presence of lexicalized negation in the verb or in adverbs;
- presence of conditional, counterfactual subordinators;
- double negations with copulative verbs;
- presence of modals and other modality operators.

It is important to remember that both Pointwise Mutual Information (PMI) and Latent Semantic Analysis (LSA) (Turney & Littman 2003) systematically omit function or stop words from their classification set of words and only consider content words. In order to cope with these linguistic elements we propose to build a propositional level analysis directly from a syntactic constituency or chunk-based representation. We implemented these additions on our system thus trying to come as close as possible to the configuration which has been used for semantic evaluation purposes in challenges like Recognizing Textual Entailment (RTE) and other semantically heavy tasks (see Bos & Delmonte 2008; Delmonte et al. 2010). The output of the system is an xml representation where each sentence of a text or dialog is a list of attribute-value pairs. In order to produce this output, the system makes use of a flat syntactic structure and a vector of semantic attributes associated to the verb compound at propositional level and memorized. An important notion required by the extraction of opinion and sentiment is also the distinction of the semantic content of each proposition into two separate categories: objective vs. subjective.

This is obtained by searching for factivity markers again at propositional level (see Saurì & Pustejovsky 2012). In particular we take into account the following markers: modality operators such as intensifiers and diminishers, modal verbs, modifiers and attributes adjuncts at sentence level, lexical type of the verb (from ItalWordNet classification, and our own), subject's person (if 3rd or not), and so on. As will become clear below, we are using a lexicon-based (see Pennebaker et al.; Taboada et al. 2011) rather than a classifier-based approach, i.e. we make a fully supervised analysis where semantic features are manually associated to lemma and concept of the domain by creating a lexicon out of frequency lists. In this way the semantically labelled lexicon is produced in an empirical manner and fits perfectly the classification needs. Now, the new current version used with Italian has been made possible by the creation of the needed semantic resources, in particular a version of SentiWordNet adapted to

Italian and heavily corrected and modified. This version uses weights for the English WordNet and the mapping of sentiment weights has been done automatically starting from the linguistic content of WordNet glosses. This process has introduced a lot of noise in the final results, with many entries with a totally wrong opinion evaluation. In addition, there was a need to characterize uniquely only those entries that have a "generic" or "commonplace" positive, or negative meaning associated to them in the specific domain. This was deemed the only possible solution to the problem of semantic ambiguity, which could only be solved by introducing a phase of Word Sense Disambiguation, which was not part of the system. However this was not possible for all entries. So, we decided to erase all entries that had multiple concepts associated to the same lemma, and had conflicting sentiment values. We also created and added an ad hoc lexicon for the majority of concepts (some 3000) contained in the texts we analysed, in order to increase the coverage of the lexicon. This was done again with the same approach, i.e. labelling only those concepts which were uniquely intended as one or the other sentiment, restricting reference to the domain of political discourse.

1.2 Semantic Mapping

Sentiment Analysis is based on propositional level semantic processing, which in turn is made of two basic components: PAS and VCX semantics. Semantic mapping is based on a number of intermediate semantic representations, which include, beside diathesis:

- Change in the World; Subjectivity and Point of View; Speech Act; Factuality; Polarity.

At first we compute Mood and Tense from the Verbal Compound (hence VC), which, as said before, may contain auxiliaries, modals, clitics, negation and possibly adverbials in between. From Mood_Tense we derive a label that is the compound tense and this is then used together with Aspectual lexical properties of the main verb to compute Change_in_the_World. Basically this results into a subclassification of events into three subclasses: Static, Gradual, Culminating. From Change_in_the_World we compute (Point_of_) View, which can be either Internal (Extensional/Intensional) or External, where Internal is again produced from a semantic labelling of the subcategorized lexicon along the lines suggested in linguistic studies, where psych(ological) verbs are separated from movement verbs etc. . Internal View then allows a labelling of the VC as Subjective for Subjectivity and otherwise, Objective. Eventually, we look for negation which can be produced by presence of a negative particle or be directly in the verb meaning as lexicalised negation. Negation, View and Semantic Class, together with presence of absence of Adverbial factual markers are then used to produce a Factuality labelling.

One important secondary effect that carries over from this local labelling, is a higher level propositional level ability to determine inferential links intervening between propositions. Whenever we detect possible dependencies between adjacent VCs we check to see whether the preceding verb belongs to the class of implicatives. We are here referring to verbs such as “refuse, reject, hamper, prevent, hinder, etc.” on the one side, and “manage, oblige, cause, provoke, etc.” on the other (for a complete list see Saurì & Pustejovsky 2012). In the first case, the implication is that the action described in the complement clause is not factual, as for instance in “John refused to drive to Boston”, from which we know that “John did not drive to Boston”. In the second case, the opposite will apply, as in “John managed to drive to Boston”.

Two notions have been highlighted in the literature on discourse: foreground and background. The foreground is that part of a discourse which provides the main information; in a narrative, for example, the foreground is the temporal sequence of events; foreground information, then, moves the story forward. The background, on the contrary, provides supportive information, such as elaborations, comments, etc., and does not move the story forward. To compute foreground and background information, three main rhetorical relations are assigned by the algorithm (for a deeper description see Delmonte 2007; 2009) in the form of attribute-value pairs, or features: Discourse Domain, CHANGE IN THE WORLD.

The Discourse Domain of a sentence may be “subjective”, indicating that the event or state takes place in the mind of the participant argument of the predicate and not necessarily in the external world. Then it may be “objective”, which indicates that the action described by the verb affects the whole environment. A sentence may also describe a “change in the world”, in case we pass from the description of one situation to the description of another situation which precedes or follows the former in time but which is not temporally equivalent to it; we have then the following inventory of changes: null (i.e. no change), gradual, culminated, earlier, negated. The third value, the “relevance” of a sentence, corresponds to the distinction between foreground and background which has been discussed above.

We have now to explain the way each utterance receives its set of values: the algorithm relies heavily on grammatical cues, i.e. those linguistic elements encoded in the grammar of a language which allow interpretation without the intervention of pragmatic or non-linguistic elements such as conversational implicatures, presupposition or inferencing. The cues we make use of are chiefly extracted from the verb and are such things as semantic category, polarity, tense, aspect. The procedure is very simple from a theoretical point of view: once the algorithm has recognized a cue, it assigns a value to the sentence. Note that we distinguish between the direct and indirect speech portions of the text, since the perspective is not the same in the two cases.

- DISCOURSE DOMAIN: to assign the point of view of a sentence, the algorithm checks the sem(antic)_cat(egory) of the main verb of the sentence and a number of other opacity operators, like the presence of future tense, a question or an exclamative, the presence of modals, etc.
- CHANGE IN THE WORLD: to establish whether a clause describes a change or not, and which type of change it describes, the algorithm takes into account four parameters: polarity (i.e. affirmative or negative), domain, tense and aspect of the main verb.

If polarity is set to NO (i.e. if the clause is negative), CHANGE is negated; but if the verb describes a state, CHANGE is null because a stative verb can never express a change, apart from the fact that it is affirmed or negated. Thus, if DISCOURSE DOMAIN is subjective and the verb is stative, CHANGE is null: this captures the fact that, in such a case, the action affects only the subject's mind and has no effects on the outside world. In all other cases the algorithm takes into account tense and aspect of the main verb and obeys the following rules: if tense is simple present, CHANGE is null; if tense is *passato remoto* or simple past, CHANGE is culminated; if tense is pluperfect or *trapassato remoto*, CHANGE is earlier; if tense is the *imperfetto* and describes a state, CHANGE is null, but if it describes an activity, a process, an accomplishment, or if it is a mental activity, CHANGE is gradual.

- FACTIVITY: this relation may only assume two values: factive and non-factive. A factive relation is assigned every time change is non null. Other sources of information may be used to trigger factivity, and that is the presence of a factive predicate, like a presuppositional verb, "know".

We now turn to the cues for direct speech. Once the algorithm has recognized a clause to be in direct speech, the CLAUSE TYPE value is dir_speech/prop. The DISCOURSE DOMAIN is also subjective: this is so because direct speech reports the thoughts and perceptions of the characters in the story, so that any intervention of the writer is left out. As far as CHANGE is concerned, the algorithm obeys the following rules: if the main verb is in the imperative mood, CHANGE is null because, although the imperative is used to express commands, there is no certainty that once a command has been imparted it is going to be carried out. If the verb is in the indicative mood, and it is in the future, CHANGE is null as well since the action has still to take place; if we have a past tense such as the *passato prossimo* or the *trapassato*, CHANGE is culminated or earlier, respectively; if tense is present, the algorithm checks its aspect: if the verb describes a state, CHANGE is null, otherwise (i.e. if the verb describes an activity) CHANGE is gradual. Finally, negative and positive polarity is carefully weighted in case the sentence has a complex structure, taking care of cases of double negations. Positives are so marked when the words searched in the input sentence belong to the class of so-called "Absolute Positives", i.e. words that can only take on positive evaluative meaning. The same applies for Negative polarity words, when they belong to a list of "Absolute Negatives", like swear words.

2. Results and Discussion

Here below is the table of our results for the three tasks of Sentipolc (see Basile et al. 2014).

Task	F-ScoreTot	Prec0	Rec0	F-score0	Prec1	Rec1	F-score1	Rank
Subjectivity	52.24	34.79	30.26	32.37	75.71	68.83	72.11	9th/9
Polarity Pos	51.81	72.97	***81.58***	77.03	43.13	16.05	23.39	10th/11
Polarity Neg	51.81	60.97	***77.00***	68.05	62.03	28.19	38.77	10th/11
Irony	49.29	88.29	77.54	82.57	15.66	16.39	16.02	4th/7

Table 2. Results of ITGetaruns for all Tasks.

In Table 2. we report percent values of our system performance. In a final column we registered our placement in the graded scale of final results. As can be noticed, best result has been achieved for irony detection. In general, we can note the following: there has been always an attempt to favour Recall rather than Precision, and also an attempt to reduce False Positives. This would be represented by a better scoring in those values associated to Prec0, Rec0 and F-score0: as can be noticed, this is only partially true. Both Polarity and Irony have by far better scoring in 0s than in 1s. On the contrary, Subjectivity has much better scores in 1s than in 0s. We assume that this is due to annotation criteria, which don't match our linguistic rules. We marked with bold italics those scores that have better ranking individually, and both coincide with Recall0 in Polarity. Recall0 for Polarity Pos is 81.58, which corresponds to the 4th rank in the list of 12 (not considering the baseline); Recall0 for Polarity Neg is 77.00, which represents the best result of all systems. Going back to annotation criteria, one of our basic rule for Subjectivity matching is presence of 1st and 2nd person morphology in the main verb complex associated to the main or root clause. We noticed that this does not always coincide with annotations associated to the tweets.

We had a number of additional features to implement, which would have increased Precision quite significantly but would have decreased Recall dramatically. One of these features was the possibility to highlight the use of alterations in Ironic tweets, which are used to express "Exaggeration". The algorithm was based on our Morphological Analyser that in turn is based on linguistic rules for alterations and a root lexicon of Italian made up of some 90,000 entries (see Delmonte, Pianta 1996; 1998). We also intended to use our classification of Emoticons, which however proved not to be a significant contribution in the overall evaluation, so at the end we decided not to implement it. Eventually, we sieved unallowed combinations of 0-1 and replaced the unwanted 1 with a zero.

As a conclusion, we intend to implement those techniques that seemed promising but required deeper inspection and were more time-consuming, like using Emoticons and alterations to detect exaggerations in tweets. This will need to make use of Predicate-Argument Structures in the hope to improve irony detection (but see Reyes & Rosso 2013). By knowing, for instance, that swear words - or exaggerations - are being using in a political context, will constitute a good hint if arguments are properly under control.

References

Basile, V., Bolioli, A., Nissim, M., Patti, V., Rosso, P. 2014. Overview of the Evalita 2014 SENTIment POLarity Classification Task, Proceedings of EVALITA'14, Pisa.

Bos, Johan & Delmonte, Rodolfo (eds.) 2008. “Semantics in Text Processing (STEP), Research in Computational Semantics”, Vol.1, College Publications, London.

Delmonte R., 2009. Computational Linguistic Text Processing - Lexicon, Grammar, Parsing and Anaphora Resolution, Nova Science Publishers, New York.

Delmonte, R. 2007. Computational Linguistic Text Processing – Logical Form, Logical Form, Semantic Interpretation, Discourse Relations and Question Answering, Nova Science Publishers, New York.

Delmonte, R., Tonelli, S., Tripodi, R. 2010. Semantic Processing for Text Entailment with VENSES, published at http://www.nist.gov/tac/publications/2009/papers.html in TAC 2009 Proceedings Papers.

Delmonte, R. (2009). Computational Linguistic Text Processing – Lexicon, Grammar, Parsing and Anaphora Resolution, Nova Science Publishers, New York.

Delmonte R. and Vincenzo Pallotta, 2011. Opinion Mining and Sentiment Analysis Need Text Understanding, in "Advances in Distributed Agent-based Retrieval Tools", “Advances in Intelligent and Soft Computing”, Springer, 81-96.

Esuli, A. and F. Sebastiani 2006. SentiWordnet: a publicly available lexical resource for opinion mining. In Proceedings of the 5th Conference on Language Resources and Evaluation LREC, 6.

Kim, S.-M. and E. Hovy, 2004. Determining the sentiment of opinions. In Proceedings of the 20th international conference on computational linguistics (COLING 2004), 1367–1373.

Pang, B. and L. Lee, 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42nd annual meeting of the Association for Computational Linguistics (ACL), 271–278.

Polanyi, Livia and Zaenen, Annie 2006. “Contextual valence shifters”. In Janyce Wiebe, editor, Computing Attitude and Affect in Text: Theory and Applications. Springer, Dordrecht, 1–10.

Reyes A., Rosso P. 2013. On the Difficulty of Automatically Detecting Irony: Beyond a Simple Case of Negation. In: Knowledge and Information Systems.

Saurì R., Pustejovsky, J., 2012. “Are You Sure That This Happened? Assessing the Factuality Degree of Events in Text”, Computational Linguistics, 38, 2, 261-299.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K. & Stede, M. 2011. “Lexicon-based methods for sentiment analysis”. In Computational Linguistics 37(2): 267-307.

Turney, P.D. and M.L. Littman, 2003. Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems (TOIS), 15–346.

Wiebe, Janyce, Wilson, Theresa, Cardie, Claire 2005. “Annotating expressions of opinions and emotions in language”. In Language Resources and Evaluation, 39(2): 165–210.