International Journal of Computational Linguistics and Applications, vol. 5, no. 2, 2014, pp. 135–152
Received 25/06/2014, accepted 22/09/2014, final 12/10/2014.
ISSN 0976-0962, http://ijcla.bahripublications.com

Implicit Aspect Indicator Extraction for Aspect-based Opinion Mining

IVAN CRUZ,[1] ALEXANDER GELBUKH,[1] AND GRIGORI SIDOROV[1]

Instituto Politécnico Nacional, Mexico

ABSTRACT

Aspect-based opinion mining aims to model relations between the polarity of a document and its opinion targets, or aspects. While explicit aspect extraction has been widely researched, limited work has been done on extracting implicit aspects. An implicit aspect is the opinion target that is not explicitly specified in the text. E.g., the sentence "This camera is sleek and very affordable" gives an opinion on the aspects appearance *and* price*, as suggested by the words "sleek" and "affordable"; we call such words Implicit Aspect Indicators (IAI). In this paper, we propose a novel method for extracting such IAI using Conditional Random Fields and show that our method significantly outperforms existing approaches. As a part of this effort, we developed a corpus for IAI extraction by manually labeling IAI and their corresponding aspects in a well-known opinion-mining corpus. To the best of our knowledge, our corpus is the first publicly available resource that specifies implicit aspects along with their indicators.*

KEYWORDS: *Aspect-based opinion mining, sentiment analysis, conditional random fields.*

1 INTRODUCTION

Opinion mining comprises a set of technologies for extracting and summarizing opinions expressed in web-based user-generated contents. It improves the quality of life for ordinary people by permitting them to consider the collective opinion of other users on a product, political figure, tourist destination, etc. It improves the incomes of businesses by letting them know what the consumers like and what they do not like. It improves the democracy by permitting political parties and governments evaluate in real time social acceptance of their programs and actions.

Opinion mining depends on accurate detection of opinions expressed in individual documents, such as blog posts, tweets, or user-contributed comments. Such detection can be done at different levels of granularity. For example, the polarity of the whole document can be determined: whether the author expresses a positive or negative opinion. For a comment on a specific product, this level of granularity might be enough. However, it is often desirable to determine sentence per sentence a specific aspect of the product on which opinion is expressed in the given sentence.

Aspect-based Opinion Mining [1, 2] considers relations between the aspects of the object of the opinion and the document polarity (positive or negative feeling expressed in the opinion). Aspect are also called opinion targets. An aspect is a concept on which the author expresses their opinion in the document. Consider, for example, a sentence “The optics of this camera is very good and the battery life is excellent.” We can say that the polarity of this review of a photo camera is positive. However, more specifically, what the author likes are *optics* and *battery life* of this camera. These concepts are the aspects of this opinion.

Aspect Extraction is the task of identifying the aspects, or opinion targets, or a given opinionated document. The aspects can be of two types: explicit aspects and implicit aspects. Explicit aspects correspond to specific words in the document: in our example, the opinion targets *optics* and *battery life* explicitly appear in the document. In contrast, an implicit aspect is not specified explicitly in the document. Consider the sentence “This phone is inexpensive and beautiful.” This sentence expresses a positive opinion on *price* and *appearance* of the *phone*. These aspects would be explicit in an equivalent sentence “The price of this phone is low and its appearance is beautiful.”

While there are many works devoted to the explicit aspect extraction, implicit aspect extraction is much less studied. Implicit aspect extraction is much more complicated than explicit aspect extraction. However, implicit aspects are ubiquitous in the documents, as the following example from the corpus described in [1] shows: *This is the best phone one could have. It has all the features one would need in a cellphone: It is lightweight, sleek and attractive. I found it very user-friendly and easy to manipulate; very convenient to scroll in menu etc.* In this example, the expressions “lightweight,” “sleek” and “attractive,” “user-friendly” and “to scroll in menu,” and “easy to manipulate” correspond to the aspects *weight*, *appearance*, *interface*, and *functionality* of the phone, correspondingly. The latter expression can be also interpreted as referring, more specifically, to the aspect *menu* of the phone. While these concepts are not explicitly mentioned in the text, they are introduced implicitly by the words that are present. We call such words, which are clues to infer the implict aspects of the opinion, *Implicit Aspect Indicators* (IAI).

Note that in this paper, we do not consider any noun as an aspect; instead, we assume that there is a pre-defined set of *aspects* (variables) of which IAI indicate the *values*. IAI differ from *implicit aspect expressions* defined by Liu [3] as “aspect expressions that are not nouns or noun phrases” in that IAI semantically refer to the values of the pre-defined aspects, irrespectively of their own surface part of speech; below we give examples of IAI expressed by nouns and noun phrases; see also Table 3.

The task of identification of implicit aspects, or implicit aspect extraction, is usually done in two phases. First, the IAI are identified in the document, e.g., “user-friendly.” Next, they are mapped to the corresponding aspects, e.g., *interface*. In this paper, we concentrate on the first step: identification of the IAI, a task that we call *implicit aspect indicator extraction*, or IAI extraction. Existing approaches to the second step (mapping IAI to aspects) are mentioned in Section 2.

An IAI could be a single word, such as “sleek,” a compound, such as “user-friendly,” or even a complete phrase, such as “to scroll in menu” in the above example.

IAI can be of different parts of speech: in “This MP3 player is really expensive,” the IAI “expensive” suggesting the aspect *price* is an adjective; in “This camera looks great,” the IAI “look” suggesting *appearance* is a verb; in “I hate this phone. It only lasted less than six months!,” the IAI “lasted” suggesting *durability* of the phone is a verb.

The following examples shows IAI as nouns or noun phrases: in “Even if I had paid full price I would have considered this phone a good deal” the IAI “good deal” suggest the aspect *price*; in “Not to mention the sleekness of this phone” the IAI “sleekness” suggest the aspect *appereance*; in “The player keeps giving random errors” the IAI “random errors” suggest the aspect *quality*; in “This phone is a piece of crap” the IAI “piece of crap” suggest the aspect *quality*.

Different IAI can correspond to the same implicit aspect. Such IAI can refer to different values of this aspect, e.g., “beautiful” or “ugly” for *appearance*, or to the same value, in which case they can be approximately synonymous, e.g., "beautiful," "pleasant," or "sleek," or participate in approximately synonymous expressions, e.g., "it is pleasant to look at this phone" or "the designer showed a very good taste."

Many authors consider only polarity or sentiment words as possible IAI. For instance, in the sentence "this phone is beautiful" the word "beautiful" has positive polarity, so it is natural to assume that indicates an opinion about some aspect, which in this case is *appearance*. Note that here the assumption is that both the aspect and the value are expressed cumulatively by the same word. While such an approach works in many cases, it fails in other cases. For example, in the sentence "the designers of this camera did a very good job," the word "designers" is not a sentiment word, but still implies the aspect *appearance*, which is not be implied by the only polarity word "good" in this sentence. Namely, here the implicit aspect is indicated by one word and its value by another word. The IAI and the word that gives its value can even appear in different sentences, e.g., "I love this phone. It works even in areas with very low signal," where "love" gives the value of the aspect *reception*.

It is not always trivial to decide whether a value of an aspect implies positive or negative opinion. For example, "the phone is very heavy" vs. "the battery lasts a lot": it is common sense that high weight for a phone is bad and high capacity for a battery is good. This is called *desirable facts*: even if the text does not contain an explicit opinion about the aspect to be good or bad but only communicates an objective fact about it, the fact should still be desirable, which implies a positive opinion, or undesirable, which implies a negative opinion. Another example: "The phone has the latest version of Android" is an objective fact, and there are no opinion words in this text; however, for a phone to have the latest version of the operating system is desirable and thus the opinion implied by it about the aspect *operating system* is positive.

In this paper, we present a novel method for IAI extraction. We use a supervised learning approach, based on sequential labeling with Conditional Random Fields (CRF). Our results show that our approach outperforms existing approaches.

To the best of our knowledge, there is no corpus for the IAI extraction task. Thus we developed such a corpus. For this, we manually labeled the IAI and their corresponding aspects in a well-known corpus for opinion mining [1]. The corpus is publicly available for research purposes.[1]

[1] Available on www.gelbukh.com/resources/implicit-aspect-extraction-corpus, visited on November 10, 2014.

The paper is organized as follows. Section 2 discusses the related work. Section 3 presents the scheme and the features we used. Section 4 describes our experimental methodology. The results are given in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

Hu and Liu [1] were the first to introduce the notion of aspect extraction in the context of opinion mining, as well as to differentiate the explicit and implicit aspects. In their work, however, they addressed only explicit aspects (using statistical rules) and did not consider any treatment of implicit aspects. Later, Popescu and Etzioni [4] and by Blair-Goldensonh [5] further improved their method.

Currently, there exist a number of methods for aspect extraction. In this paper we will present a method based on a supervised learning technique. Thus, in the rest of this section we will focus on the supervised learning methods.

The task of aspect extraction is a particular case of information extraction task. There exist various methods for the latter task [6, 7], of which the most dominant ones are based on *sequential labeling*. There are two main techniques for sequential labeling: Hidden Markov Models (HMM) and Conditional Random Fields [8] (CRF).

Various methods have been applied for aspect extraction. Lexicalized HMM were applied to extract the opinions paired with the corresponding explicit aspects [9]. CRF were used by various authors for explicit aspect extraction [10–13].

Fewer works addressed implicit aspect extraction. The first system of this kind, OPINE [4], was introduced in order to achieve better polarity classification. Unfortunately, this system is not well-documented and not available for public.

All methods for implicit aspect extraction we are aware of rely on what we in this paper call IAI. In all works, only sentiment words are considered as candidates for IAI. Clustering was used to convert such IAI into explicit aspects, basing on the statistics of co-occurrence of explicit aspects and sentiment words in the sentences [14]. Two-phase co-occurrence association rule mining was used to relate implicit and explicit aspects [15]. In another rule-based method, explicit aspects were identified in the text and then implicit aspects were mapped to them by clustering the pairs of explicit aspects and sentiment words that were candidates to implicit aspects [16].

Recently, rule-based frameworks has shown very promising results for extraction of implicit and explicit aspects [17] and for aspect-based sentiment analysis [18, 19], especially by using concepts and not single words [20]. Identification of sentiment words and sentiment orientation of the text is in turn a task that has been dealt with using rule-based approaches [21], machine-learning methods [22–24], and lexical resources [25, 26].

3 Methodology

In what follows we describe the scheme used for IAI extraction and the features we used during our experiments.

3.1 *IAI Extraction*

The objective is to label words from an opinionated input text as IAI. Figure 1 shows an example. There is an opinionated sentence as input. The output is a set of duples. Each duple consists of a token of the sentence and the label of a class assigned by an IAI extraction method. The label 'I' is for the class "IAI" and the label 'O' is for the class "Other." The words "sleek" and "affordable" are classified as IAI.

INPUT: "*This phone is sleek and very affordable.*"
OUTPUT: {('*This*',O),('*phone*',O),('*is*',O),
('*sleek*',I),('*and*',O),('*very*',O),
('*affordable*',I),('.',O)}

Fig. 1. IAI extraction example

We cast the task of IAI extraction as a sequence labelling task. Let $X = \{x_1, ..., x_m\}$ be a set of observations and $Y = \{y_1, ..., y_m\}$ a set of assigned labels to those observations. The objective is to predict the set of labels $Y' = \{y_{m+1}, ..., y_n\}$ given a set of new inputs $X' = \{x_{m+1}, ..., x_n\}$ with a model obtained with the observed data X and the given labels Y. The sequential labelling method used is Conditional Random Fields.

3.2 *Conditional Random Fields*

Conditional Random Fields (CRF) is a probabilistic graphical framework for building probabilistic models to segment and label sequences of data. It takes a discriminative approach. More generally, a CRF is a log-linear model that defines a probability distribution over sequences of data given a particular observation sequence. Lafferty et al. [8] defined a CRF on a set of observations X and a set of label sequences Y as follows: Let $G = (V, E)$ be a graph such that $Y = (Y_v)_{v \epsilon V}$ so that Y is indexed by the vertices of G, then (X, Y) is a conditional random field in case, when conditioned on X the random variables Y_v, obey the Markov property with respect to the graph:

$$p(Y_v|X, Y_u, u \neq v) = p(Y_v|X, Y_u, u \sim v), \quad (1)$$

where $u \sim v$ means that w and v are neighbors in G. This property describes the fact that the conditional probability of a label Y_v depends only on a label Y_u iff there is *affinity* with Y_v, i.e. $(Y_v, Y_u) \epsilon E$.

The joint distribution over the label sequences Y given X has the form:

$$p_\theta(y|x) \propto exp\left(\sum_{e \epsilon E,k} \lambda_k f_k(y|_e, x) + \sum_{v \epsilon V,k} \mu_k g_k(v, y|_v, x)\right), \quad (2)$$

where x is the data sequence, y is a label sequence, $y|_S$ is the set of components of y associated with the vertices in subgraph S, f_k and g_k are *feature functions* and θ is the set weight parameters

$$\theta = (\lambda_1, \lambda_2, \lambda_3, ...; \mu_1, \mu_2, \mu_3, ...).$$

The feature functions f_k and g_k are a set of functions that maps a set of observations X to a real number, typically to the subset $\{0, 1\}$. These functions are built in order to model an observation X_i as a vector. We assume that the features are given and fixed. They are usually Boolean and crafted by hand. For example, a vertex feature f_k can be true (i.e, f_k maps the observation X_i to 1) if the word X_i is upper case and the tag Y_i is proper noun.

For our proposed approach we used a particular case of this framework: Linear Chain Conditional Random Fields Sequence Labeling [8]. This is a supervised method for predicting label sequences given a set of observations. The implementation used in our experiment was the CRFClassifier included in the Stanford NER[2] [27]. This classifier is a Java implementation of arbitrary-order linear-chain CRF sequence models.

3.3 *Features*

The data used for training the CRF-based labeller was taken from the dataset described in Section 4.1. We pre-processed the data by removing punctuation and stop words. Capitalized and upper-case words were left as they are.

The Stanford NER includes a Java class named NERFeatureFactory. This class implements several feature extraction methods. One can enable (with a configuration file) specific feature extractors to use them with the CRFClassifier in order to build a feature vector. We used this class to built such feature vectors for our experiments.

Given a sequence of words, we construct a feature vector for each word to be labelled. These feature vectors contains the following features encoded:

1. Word Features: These are features that indicate which word type is the actual instance to be labelled.
2. Character n-grams features: These are features that indicate if a substring appears in a word. These type of features have been proved useful in Name Entity Recognition tasks [28]. The substrings are from the corpus types. A restriction on these n-grams is that they are not be larger than 6 characters. This restriction is because with larger n-grams the training becomes very expensive in terms of computational power, with little classification performance gain. Other restriction is that they do not contain either the beginning or end of the word. We determined experimentally that n-grams with these properties give better performance.
3. Part of Speech (POS) tag features: The POS tag of the word. For these features, one must provide the POS tag for each token in a sentence as input. We used the NLTK POS Tagger[3] for tagging.
4. Context Features: These are the word, tag and the combination word-POS tag of the previous and next word of the current instance to be labelled.

[2] http://nlp.stanford.edu/software/CRF-NER.shtml

[3] http://nltk.org/

5. Class sequences features: These are the combination of the given particular word with the labels given to the previous words. We used a label window of 2, i.e. the labels of the 2 previous words plus the current word and as features.
6. Word bi-gram features.

4 EXPERIMENTAL SETUP

The general description of our experimental setup is as follows: first we developed a corpus for IAI extraction, then we defined the different metrics and validation methods for our experiments. Finally, we defined the baselines to compare the performance of our approach.

4.1 *Dataset*

We noticed that there was no suitable dataset for our experiments. As explained in Section 1, limited work has been done in extracting implicit aspects. Moreover, the task that we call IAI extraction was not defined since the common approach to infer implicit aspects was to take sentiment words as the best words to infer such aspects. Therefore, it is natural that there are no resources for IAI extraction (as far as we know). As a result of this, we developed the first corpus for IAI extraction.

Hu and Liu [1] developed a corpus for explicit aspect extraction. This corpus has been widely used in many opinion mining subtasks. We used the texts of this corpus to create a new one for IAI extraction. We labelled the text indicating the IAI and their corresponding implicit aspects. We only selected sentences that have at least one implicit aspect in order to label the corpus. Therefore, we did not label every opinionated sentence.

Table 1 shows some of the properties of the IAI corpus. It consists of 314 Amazon reviews of 5 products in the electronics commodities domain: a DVD player (the column “DVD” in the table), a Canon camera (“Canon”), an MP3 player (“MP3”), a Nikon camera (“Nikon”) and a Nokia Cellphone (“Phone”). This table describes the number of reviews per document. It also describes how many words and sentences a review has on average.

The corpus statistical properties at different granularity levels are shown in Table 2. The the name of each column is the same as the name of the columns in Table 1. This table is divided in 3 section for each granularity level:

– Sentence level.
– Token level.
– Type level.

The sentence-level section shows how many sentences are in the document, how many sentences of the documents have at least one IAI (shown in the row labelled as "IAI#") and the percentage of sentences that have at least one IAI ("IAI%"). The token-level and type-level section describes the same properties for these granularity levels.

Table 1. Corpus Properties.

	DVD	Canon	MP3	Nikon	Phone
Reviews	99	45	95	34	41
Words per Review	572.3	1236.4	1575.5	924	1085.3
Sentences per Review	7.47	13.26	18.90	3.64	13.31

Table 2. Statistical Properties.

	DVD	Canon	MP3	Nikon	Phone
Sentence level					
Sentences	740	597	1796	346	546
IAI#	147	63	155	36	44
IAI%	19.86%	10.55%	9.03%	10.40%	8.05%
Token level					
Tokens	56661	55638	149676	31416	44497
IAI#	164	79	214	50	66
IAI%	0.289%	0.141%	0.142%	0.159%	0.148%
Type level					
Types	1767	1881	3143	1285	1619
IAI#	72	63	136	40	42
IAI%	4.07%	3.34%	4.32%	3.11%	2.59%

The POS distribution for the IAI labeled in the corpus is shown in Table 3. Each row represents a general Penn Treebank POS tag. The first row represents all the tags that are adjectives (JJ, JJR, JJS), the second one represents the noun tags (NN,NNS, NNP, NNPS) and the third one represents the verb tags (VB, VBD, VBG, VBN, VBP, VBZ). The last row is the rest of tags seen with an IAI. The IAI column shows how many vocabulary words were seen labeled with the given tag. The third column describes how many words with the given tag were seen in sentences with IAI. The fourth column shows the tag distribution observed in the IAI.

Table 3. Corpus POS distribution

POS	IAI	POS in IAI Sentence	P(IAI)
JJ	157	527	0.2818
NN	167	1692	0.3000
VB	220	1112	0.3836
other	19	3900	0.0346

4.2 *Metrics and Validation Methods*

We used our annotated corpus as gold standard. The labeled IAI include compounds and phrases. Labeled words as IAI must match those labeled as IAI in the corpus, which are counted as true positives (tp). Those words that do not match are counted as false positives (fp). False negatives (fn) are words labeled as IAI in the corpus that were not extracted as IAI.

We measured the precision and recall. Precision P and recall R are defined as

$$P = \frac{tp}{tp + fp}, \qquad R = \frac{tp}{tp + fn}.$$

The performance metric used was the $F1$ Score. It is defined as

$$F1 = 2 \cdot \frac{P \cdot R}{P + R}.$$

The results were obtained in a 10-fold cross validation setup.

4.3 *Baseline Approach*

The first baseline is to label the sentiment words as IAI for each sentence in a review [14–16]. We call this baseline BSLN1. We use the sentiment lexicon used in [2] to determine the opinion polarity of words. This lexicon is conformed by two word lists. The first list is conformed by ”positive words”, which are words that suggest a positive opinion in an opinionated context (e.g ”awesome”). The second list is conformed by ”negative words”. These words suggest a negative opinion (e.g. ”awful”).

The algorithm for this baseline is as follows: for each word in a sentence we determine if this word is in any of the two list of the lexicon. If it is, we label it as IAI.

We propose a second baseline based on text classification, which we call BSLN2: we implemented a Naive Bayes (NB) text classifier. The classifier was trained with the texts of our developed corpus. The task of this classifier is to determine whether a sentence has at least one IAI or not. If a sentence is classified as a one with IAI, we label the sentiment words as IAI.

The features used in the NB classifier were:

- Corpus vocabulary stems. We exclude stop words.
- The best 500 bi-gram collocations obtained by a Point-wise Mutual Information association measure.

Finally, we also implemented a second-order Hidden Markov Model sequence labeller. This is the standard method for sequence labelling. We called this method BSLN3. We trained this labeller with our corpus. Since the labeller is a second order HMM, we use bigrams and trigrams as features. The training data is pre-processed as follows:

- The words that appear fewer than 5 times in the corpus (rare words) are changed in the training data for the label *RARE*.
- The rare words that contain at least one numeric character are changed for the label *NUMERIC*
- The rare words that consist entirely of capitalized letters are changed for the label *ALLCAPS*

All baselines were implemented in Python. We used the NB Classifier included in NLTK for the BSLN2.

5 RESULTS

Table 4 shows the performance of BSLN2 classifying sentences with at least an IAI within.

Table 5 compares the performance of the baselines and our CRF-based approach with different features combinations. We call *WT* the

Table 4. BSLN2 sentences classification performance

	Precision	Recall	F1-Score
Extraction of Sentence with IAI	0.25	0.37	0.30

combination of the word and tag features (points 1 and 3 from the features description in Section 3.3). *CNG* features are the character n-grams features (point 2). *CNTX* are context features and word bigram features (points 4 and 6). *CLS* are the class sequence features (point 5).

We observed that the WT features give the greatest precision. However the recall is poor.

The CNG features give a recall boost. These features capture the morphological properties of the words (roots, prefixes, suffixes). Words with similar morphological properties tend to be semantically similar. For example the sentence “This phone looks great” could be rephrased as “The phone’s look is great” or even “This phone looked great with its case.” The root “look” present in the previous sentences is the best IAI to infer the *appearance* aspect. Therefore words with this root should have greater probability of being extracted as IAI. The drawback is that these features decrement the overall precision because more words that are not IAI but contain these character n-grams will have greater probability of being extracted.

The CNTX and CLS features improve both precision and recall. The best performance is obtained with the combination of WT, CNG, CNTX and CLS features.

Table 5. IAI Extraction performance with different features

	Precision	Recall	F1 Score
BSLN1	0.0381	**0.3158**	0.0681
BSLN2	0.1016	0.1379	0.1170
BSLN3	0.5307	0.1439	0.2264
WT	**0.6271**	0.0575	0.1053
CNG	0.4765	0.1925	0.2742
CNTX	0.5030	0.1148	0.1869
WT,CNG	0.4697	0.1992	0.2795
WT,CNG,CNTX	0.5209	0.2031	0.2932
WT,CNG,CNTX,CLS	0.5458	0.2064	**0.2970**

Our approach is better for this task. It outperforms significantly both BSLN1 and BSLN2. The precision is very high. However the recall is lower than BSLN1.

In order to tradeoff precision for recall in our CRF-based approach, we used a biased CRF classifier [29]. This method allows to set a bias towards the different classes. These biases (which internally are treated as feature weights in the log-linear model underpinning the CRF classifier) can take any real value. As the bias of a class *A* tends to plus infinity, the classifier will only predict *A* labels, and as it tends towards minus infinity, it will never predict *A* labels. These biases are used to manually adjust the precision-recall tradeoff.

We experimented with the *IAI* class bias. We changed the value of this bias within a range of 1.5 to 3.5. We keep the *Other* class bias value fixed to 1. The set of features used are those shown in the last row of Table 5. Table 6 shows the precision, recall and F1 Score of several experiments with different IAI class bias values. Figure 2 shows a graphic of this data.

Table 6. Precision, Recall and F1 Score with different IAI Class bias

IAI Class Bias	Precision	Recall	F1 Score
1.5	**0.5252**	0.2602	0.3479
2.0	0.4636	0.3095	0.3711
2.5	0.4201	0.3503	**0.3820**
3.0	0.3656	0.3850	0.3750
3.5	0.3184	**0.4203**	0.3623

The best F1 Score is obtained with an IAI class bias value of 2.5. It gives a boost of 28.61% in terms of the IAI extraction performance without IAI class bias. Furthermore both precision and recall are higher than any of the baselines.

6 CONCLUSIONS

We have described a model for extracting what we call Implicit Aspects Indicators, which are words that infer implicit aspects of an opinionated document using Conditional Random Fields. We developed a dataset for this task based on a well-know corpus for opinion mining. Also we presented a comparative performance evaluation of our approach with three baselines. The results shown that our approach outperforms significantly

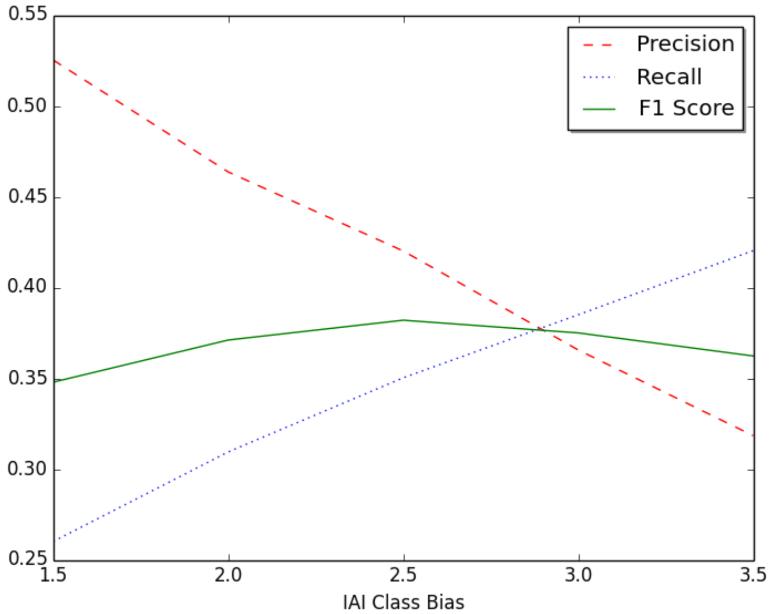

Fig. 2. Precision, Recall and F1 Score with Biased CRF

these baselines. The features used were described and we shown they are not complicated yet quite effective in IAI extraction.

For future work we are going to study new features for this task. We believe that syntactic dependency features could improve the performance [30, 31]. Finally we are working on a Implicit Aspects extraction model based on IAI. We will explore several approaches for mapping IAI with implicit aspects using semantic similarity [32–35].

REFERENCES

1. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. (Aug 2004) 168–177
2. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: Proceedings of First ACM International Conference on Web Search and Data Mining (WSDM-2008), Stanford University, Stanford, California, USA (Feb 2008) 231–240

3. Liu, B.: Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers (2012)
4. Popescu, A.M., Etzioni, O.: Extracting product features and opinions from reviews. In: Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2005). (2005) 3–28
5. Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G.A., Reynar, J.: Building a sentiment summarizer for local service reviews. In: Proceedings of WWW-2008 Workshop on NLP in the Information Explosion Era. (2008) 14
6. Hobbs, J.R., Riloff, E.: Information extraction. In Indurkhya, N., Damerau, F., eds.: Handbook of Natural Language Processing. Chapman & Hall/CRC Press (2010) 511–532
7. Mooney, R.J., Bunescu, R.: Mining knowledge from text using information extraction. ACM SIGKDD Explorations Newsletter **7(1)** (2005) 3–10
8. Lafferty, J., McCallum, A., Pereira, F.C.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the 18th International Conference on Machine Learning, Morgan Kaufmann Publishers (2001) 282–289
9. Jin, W., Ho, H.H.: A novel lexicalized HMM-based learning framework for web opinion mining. In: Proceedings of International Conference on Machine Learning (ICML-2009). (2009) 465–472
10. Niklas, J., Gurevych, I.: Extracting opinion targets in a single and cross-domain setting with conditional random fields. In: Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2010). (2010) 1035–1045
11. Li, F., Han, C., Huang, M., Zhu, X., Xia, Y.J., Zhang, S., Yu, H.: Structure-aware review mining and summarization. In: Proceedings of the 23rd International Conference on Computational Linguistics (COLING-2010). (2010) 653–661
12. Choi, Y., Cardie, C.: Hierarchical sequential learning for extracting opinions and their attributes. In: Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL-2010). (2010) 268–274
13. Huang, S., Liu, X., Peng, X., Niu, Z.: Fine-grained product features extraction and categorization in reviews opinion mining. In: Proceedings of the IEEE 12th International Conference on Data Mining Workshops. (2012) 680–686
14. Su, Q., Xu, X., Guo, H., Guo, Z., Wu, X., Zhang, X., Swen, B., Su, Z.: Hidden sentiment association in Chinese web opinion mining. In: Proceedings of International Conference on World Wide Web (WWW-2008). (2008) 959–968
15. Zhen, H., Chang, K., Kim, J.: Implicit feature identification via co-occurrence association rule mining. In: Computational Linguistics and Intelligent Text Processing. 12th International Conference, CICLing 2011, Tokyo, Japan, February 20–26, 2011. Proceedings, Part I. Volume 6608 of Lecture Notes in Computer Science. (2011) 393–404

16. Zeng, L., Li, F.: A classification-based approach for implicit feature identification. In: Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data. 12th China National Conference, CCL 2013 and First International Symposium, NLP-NABD 2013, Suzhou, China, October 10–12, 2013, Proceedings. Volume 8202 of Lecture Notes in Computer Science. (2013) 190–202
17. Poria, S., Cambria, E., Ku, L.W., Gui, C., Gelbukh, A.: A rule-based approach to aspect extraction from product reviews. Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP) (2014) 28–37
18. Poria, S., Ofek, N., Gelbukh, A., Hussain, A., Rokach, L.: Dependency tree-based rules for concept-level aspect-based sentiment analysis. In: Semantic Web Evaluation Challenge. Springer International Publishing (2014) 41–47
19. Poria, S., Gelbukh, A., Agarwal, B., Cambria, E., Howard, N.: Sentic demo: A hybrid concept-level aspect-based sentiment analysis toolkit. ESWC 2014 (2014)
20. Cambria, E., Poria, S., Gelbukh, A., Kwok, K.: A common-sense based API for concept-level sentiment analysis. Making Sense of Microposts **1**(1) (2014) 1–2
21. Poria, S., Cambria, E., Winterstein, G., Huang, G.B.: Sentic patterns: Dependency-based rules for concept-level sentiment analysis. Knowledge-Based Systems **69** (2014) 45–63
22. Poria, S., Gelbukh, A., Cambria, E., Das, D., Bandyopadhyay, S.: Enriching SenticNet polarity scores through semi-supervised fuzzy clustering. In: Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on, IEEE (2012) 709–716
23. Poria, S., Gelbukh, A., Das, D., Bandyopadhyay, S.: Fuzzy clustering for semi-supervised learning—case study: Construction of an emotion lexicon. Proceedings of MICAI (2012)
24. Das, D., Poria, S., Bandyopadhyay, S.: A classifier based approach to emotion lexicon construction. In: Natural Language Processing and Information Systems. Springer Berlin Heidelberg (2012) 320–326
25. Poria, S., Gelbukh, A., Hussain, A., Howard, N., Das, D., Bandyopadhyay, S.: Enhanced SenticNet with affective labels for concept-based opinion mining. IEEE Intelligent Systems (2) (2013) 31–38
26. Poria, S., Gelbukh, A., Cambria, E., Yang, P., Hussain, A., Durrani, T.S.: Merging SenticNet and WordNet-Affect emotion lists for sentiment analysis. In: Signal Processing (ICSP), 2012 IEEE 11th International Conference on. Volume 2., IEEE (2012) 1251–1255
27. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by Gibbs sampling. In: Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005). (2005) 363–370

28. Klein, D., Smarr, J., Nguyen, H., Manning, C.: Named entity recognition with character-level models. In: Proceedings of CoNLL-2003. (2003) 180–183
29. Minkov, E., Wang, R.C., Tomasic, A., Cohen, W.W.: NER systems that suit user's preferences: Adjusting the recall-precision trade-off for entity extraction. In: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers. (2006) 93–06
30. Sidorov, G.: N-gramas sintácticos no-continuos. Polibits **48** (2013) 69–78
31. Sidorov, G.: Should syntactic n-grams contain names of syntactic relations? International Journal of Computational Linguistics and Applications **5**(2) (2014) 17–38
32. Das, N., Ghosh, S., Gonçalves, T., Quaresma, P.: Comparison of different graph distance metrics for semantic text based classification. Polibits **49** (2014) 51–57
33. Huynh, D., Tran, D., Ma, W., Sharma, D.: Semantic similarity measure using relational and latent topic features. International Journal of Computational Linguistics and Applications **5**(1) (2014) 11–25
34. Wolf, L., Hanani, Y., Bar, K., Dershowitz, N.: Joint word2vec networks for bilingual semantic representations. International Journal of Computational Linguistics and Applications **5**(1) (2014) 27–42
35. Sidorov, G., Gelbukh, A., Gómez-Adorno, H., Pinto, D.: Soft similarity and soft cosine measure: Similarity of features in vector space model. Computación y Sistemas **18**(3) (2014) 491–504

IVAN CRUZ
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN,
INSTITUTO POLITÉCNICO NACIONAL,
AV. JUAN DE DIOS BÁTIZ, S/N, 07738, MEXICO CITY, MEXICO
E-MAIL: <ICRUZ_B12@SAGITARIO.CIC.IPN.MX>

ALEXANDER GELBUKH
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN,
INSTITUTO POLITÉCNICO NACIONAL,
AV. JUAN DE DIOS BÁTIZ, S/N, 07738, MEXICO CITY, MEXICO
E-MAIL: <WWW.GELBUKH.COM>

GRIGORI SIDOROV
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN,
INSTITUTO POLITÉCNICO NACIONAL,
AV. JUAN DE DIOS BÁTIZ, S/N, 07738, MEXICO CITY, MEXICO
E-MAIL: <SIDOROV@CIC.IPN.MX>