PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 103 APRIL 2015 21–41

Resources for Indonesian Sentiment Analysis

Franky, Ondřej Bojar, Kateřina Veselovská

Charles University in Prague, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics

Abstract

In this work, we present subjectivity lexicons of positive and negative expressions for Indonesian language created by automatically translating English lexicons. Other variations are created by intersecting or unioning them. We compare the lexicons in the task of predicting sentence polarity on a set of 446 manually annotated sentences and we also contrast the generic lexicons with a small lexicon extracted directly from the annotated sentences (in a cross-validation setting). We seek for further improvements by assigning weights to lexicon entries and by wrapping the prediction into a machine learning task with a small number of additional features. We observe that lexicons are able to reach high recall but suffer from low precision when predicting whether a sentence is evaluative (positive or negative) or not (neutral). Weighting the lexicons can improve either the recall or the precision but with a comparable decrease in the other measure.

1. Introduction

Sentiment analysis has gained much attention lately mostly due to its practical applications in commercial settings. The task is being widely solved not only for English, but also for many other languages, including languages with scarce evaluative data. However, we are not aware of any systematic attempts to build sentiment analysis resources for Indonesian so far, despite the increasing use of Internet by speakers of Indonesian language.

In this paper, we present our work on two types of resources for Indonesian sentiment analysis. The first one is a small collection of sentences coming from user reviews in several domains, manually annotated for sentiment. The second one is a collection of subjectivity (sentiment) lexicons built mainly by translating available English lexicons using several methods of translation. We use these resources together

 Corresponding author: bojar@ufal.mff.cuni.cz
Cite as: Franky, Ondřej Bojar, Kateřina Veselovská. Resources for Indonesian Sentiment Analysis. The Prague Bulletin of Mathematical Linguistics No. 103, 2015, pp. 21–41. doi: 10.1515/pralin-2015-0002.

and evaluate the performance of a simple lexicon-based sentiment analysis method on the annotated data.

2. Previous Work

For non-English languages, the creation of subjectivity lexicons usually takes the advantage of the availability of WordNet for the given language. This can be found in Bakliwal et al. (2012) and Pérez-Rosas et al. (2012). The work by Bakliwal et al. (2012) is for Hindi. They start with small seeds of 45 adjectives and 75 adverbs, pre-annotated with positive, negative, or objective polarity information. The seeds are expanded using Breadth First expansion by looking at the antonymy relation for opposite polarity and synonymy for the same polarity. Pérez-Rosas et al. (2012) in their work on Spanish subjectivity lexicon take the advantage of aligned synsets between WordNets of different languages to do the mapping. They get two different lexicons. A full strength lexicon is created by taking words with strong negative or positive polarity from MPQA[1] lexicon and map them to the synsets in SentiWordNet, by taking the synset with the highest negative or positive value for each word. The found synsets are mapped to Spanish WordNet. The second lexicon, a medium strength lexicon, is created by mapping the synsets in SentiWordNet with polarity scores greater than 0.5 to Spanish WordNet.

A subjectivity lexicon for Dutch adjectives is created by Smedt and Daelemans (2012) using a mixture of manual annotation and automatic expansion. The first step is to extract adjectives with high frequencies from a collection of book reviews. Seven human annotators annotate the adjectives that are previously disambiguated using CORNETTO (an extension of Dutch WordNet). Each adjective is expanded by their best nearest neighbours (handpicked by two annotators) from the list of new adjectives taken from the corpus and using cosine similarity as the measure of similarity. Each adjective is represented as a vector of top 2,500 nouns from the same corpus. Another expansion is performed by adding words from the same synset in CORNETTO, and by using the relations provided, e.g., antonymy, synonymy.

A method of creating a subjectivity lexicon for a language with scarce resources (Romanian) is introduced by Banea et al. (2008). They propose a method to create subjectivity lexicon using an online dictionary and a collection of documents. The work uses a set of subjective words called seed words to bootstrap the lexicon creation. The process runs by querying the online dictionary using these seed words. A list of extracted words returned by dictionary for each seed word is then filtered by calculating their similarity with the seed word using Latent Semantic Analysis (LSA). The LSA module is trained on Romanian corpus of half-million words. The surviving words are added to the lexicon and the process is repeated until the maximum number of iterations is reached.

[1] http://mpqa.cs.pitt.edu/

Some other approaches in subjectivity lexicon creation for non-English languages that do not utilize dictionary or thesaurus (e.g., WordNet) can be found in Maks and Vossen (2012) and Kaji and Kitsuregawa (2007). Their works can be considered as corpus-based approaches to lexicon creation. Maks and Vossen (2012) use an underlying assumption that different types of corpus posit different characteristics of subjectivity or objectivity information. They use three different corpora of Wikipedia articles, news, and comments inside the news to build Dutch subjectivity lexicon. They take words in the news and comments that are not over-used in Wikipedia articles as subjective words. The measures of over-usage of words between the corpus are calculated using log-likelihood ratio and a DIFF calculation (Gabrielatos and Marchi, 2011).

Kaji and Kitsuregawa (2007) exploit the dependencies and language structures in Japanese to extract evaluative sentences from a collection of one billion HTML documents. They use a list of cue words to detect the presence of evaluative clauses (positive or negative) in the dependency structure of the sentence. They also use layout structures such as itemization/table in HTML documents and cue words such as 'pros and cons' and 'plus and minus' to extract positive and negative evaluative sentences. From the evaluative sentences, they extract candidate phrases consisting of adjectives and adjective phrases, e.g. noun+adjective, together with their counts in positive and negative sentences. The candidates are then filtered using chi-square and PMI polarity score, and pre-defined thresholds.

3. Annotated Sentences

In this section, we describe our annotated data. The annotated sentences were taken from user reviews on *KitaReview* website[2]. We randomly selected 24 reviews and segmented them into separate sentences. The sentences were manually checked and cleaned, removing incomplete or otherwise broken ones. The final set consists of 446 sentences.

The annotation of the sentences was performed by two native speakers of Indonesian. The annotation process equipped the sentences with the following information:

- **Sentence objectivity/subjectivity**. Annotating the sentence as objective (o), i.e. factual, expressing no opinion, or subjective (s), i.e. expressing opinion.
- **Sentence polarity**. The overall polarity of the sentence, i.e. an estimate whether the sentence makes a positive (pos), negative (neg), or neutral (non) impression on the reader.
- **Evaluative Expressions**. The words in the sentences that are considered to bear positive or negative polarity are explicitly marked: "#expression@" for positive expressions, and "#expression$" for negative ones, e.g., "meskipun relatif

[2] http://www.kitareview.com

sedikit lebih #mahal$ (expensive) ... cukup #sepadan@ (worth) dengan segala kualitas masakan".

- **Two targets flag**. While in our limited annotation, we do not explicitly mark the target(s) of the valuation(s) expressed in the annotated sentences, it is not uncommon that a sentence attributes some valuation to more than one object. Sometimes the valuations can be even contradictory. In order to at least estimate how often this complication occurs, we explicitly mark sentences with two or more targets with the flag "_TWOTARG".

Table 1 shows basic statistics of the annotation. We see that around 60% of the sentences are marked as neutral. The number of negative sentences is much lower than that of the positive ones. It can also be observed that our annotators explicitly marked more positive expressions compared to the negative ones.

Table 1. Summary of Annotated Sentences

	Annotator 1	Annotator 2
Neutral Sentences	267	281
Positive Sentences	157	150
Negative Sentences	22	15
Sentences with Two Targets	30	17
# Pos Expressions (unique)	151	114
# Neg Expressions (unique)	40	33

The annotated set of sentences is stored in a plain text file, each sentence on a separate line. Tab-delimited columns contain all the information, as summarized in Table 2.

3.1. Agreement on Overall Polarity of a Sentence

We calculated the inter-rater agreement for overall polarity (sentiment) of the two annotations using the Kappa (κ) statistic. The agreement on the level of annotating the 446 sentences as neutral vs. evaluative (i.e. positive or negative but regardless which of these two classes) is 0.697.

If we restrict the set of sentences to the 140 ones where both annotators marked the sentence as evaluative, the agreement on the actual polarity (positive or negative) is higher: κ of 0.921.

4. Subjectivity Lexicons

To the best of our knowledge, there are no subjectivity lexicons (lists of positive or negative expressions) for Indonesian.

Table 2. Columns of Our Annotated Dataset

Column Name	Description
SENTENCEID	ID of the sentence
DOCID	Review document ID of the sentence
LINK	URL of the review on the original website
CATEGORY	Domain of the review
TITLE	Title of the review
REVIEWER	The author of the review
SENTENCE	The full text of the annotated sentence, including markup for evaluative expressions and the optional "TWOTARG" flag.
OBJSUBJ	Indication whether the sentence is objective (factual) or subjective (expressing an opinion)
POSNEGNON	The overall sentence polarity (sentiment): positive/negative/neutral

We created several such lexicons from English ones by (automatic) translation. The translated lexicons were then merged by intersecting or unioning them based on their source lexicon or the method of translation. The selection of the method of lexicon creation by translating was based on some limitations of the language resources available in Indonesian. In total, we produced 12 subjectivity lexicons from translation alone and 16 lexicons from merging operations.

4.1. Producing the Basic Lexicons

We used four different English subjectivity lexicons as our source lexicons as listed below:

- Bing Liu's Opinion Lexicon[3]
  It is a subjectivity lexicon created and maintained by Bing Liu (Hu and Liu, 2004). It is a list of around 6,800 entries (positive and negative combined).
- Harvard General Inquirer
  General Inquirer lexicon[4] is a list containing words and various syntactic and semantic features or categories. The positive and negative categories can be used to extract positive and negative words from the list.

[3] http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

[4] http://www.wjh.harvard.edu/~inquirer/spreadsheet_guide.htm

- MPQA (Multi-Perspective Question Answering) Subjectivity Lexicon
  MPQA[5] or OpinionFinder[6] lexicon is a subjectivity lexicon built using manual and automatic identification of evaluative words (Wilson et al., 2005). The lexicon contains words and information about their polarities, subjectivity strengths, and also their part-of-speech tags. We took the words with 'priorpolarity' tag of 'positive', 'negative', or 'both'. In the case of 'both', we put the word in both positive and negative lists.
- SentiWordNet[7]
  SentiWordNet (Baccianella et al., 2010) is a lexicon created on the basis of WordNet synsets (Miller, 1995) by assigning polarity weight (positive/negative) to the synsets. The general approach to produce the lexicons is by using a random walk model to propagate the positive and negative weight using the relationship information found in the gloss of the synsets. We took synsets that have polarity weight (positive/negative) greater than or equal to 0.5 ($>= 0.5$).

Table 3 below provides the exact numbers of positive and negative expressions in each of the English lexicons. We found duplicated entries in our English General Inquirer lexicon. The numbers without duplication are 1,637 for positive lexicon and 2,005 for negative lexicon. This duplication does not affect the resulting Indonesian lexicons, since we run de-duplication process before producing the final lexicons, see below.

We used three methods of translation to convert the extracted English lists of words into Indonesian:

- Google Translate[8]
  We simply copied and pasted all the entries from a list into the web interface of Google Translate. We translated one list at a time, e.g., positive list from SentiWordNet, with each entry separated by a newline. The translation was carried out during November 2012.
- Moses[9]
  We used Moses and our parallel corpus of 40,369 sentences (with no additional annotation) to build a small statistical machine translation system, with 38,369 sentences for training and 2,000 sentences for tuning, and using default parameters. The training data come from several domains: news, official reports, and devotional articles.

[5] http://mpqa.cs.pitt.edu/

[6] http://mpqa.cs.pitt.edu/opinionfinder/

[7] http://sentiwordnet.isti.cnr.it

[8] http://translate.google.com

[9] http://www.statmt.org/moses

- Kamus Online Bilingual Dictionary
  We used the online bilingual dictionary Kamus.net[10] to translate expressions from the English lexicons and took only the first translation as the result.

	Source Expressions		Covered by a Translation System					
			Google		Moses		Kamus	
Lexicon	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
Bing Liu	2,006	4,783	91%	87%	15%	6%	63%	61%
General Inquirer	1,915	2,291	99%	99%	31%	16%	82%	81%
MPQA	2,321	4,168	94%	91%	18%	8%	67%	61%
SentiWordNet	5,730	8,821	80%	73%	18%	14%	50%	41%
Intersection	470	791						
Union	7,809	12,445						

Table 3. Number of expressions extracted from English subjectivity lexicons and the extent to which they are translatable by each examined translation system.

Table 3 summarizes the coverage of each of translation systems. A term is considered non-translated if the system fails to produce any translation as well as when it copies the input verbatim to the output.

Google Translate appears to have the best coverage while our Moses (esp. due to the relatively small training data) covers the fewest items.

After the automated translation, we removed untranslated and duplicated entries. One annotator then manually checked all entries and removed translations that did not convey evaluative sense and also translations that consisted of more than one word but did not form a single multi-word expression. Table 4 shows the number of expressions for each lexicon produced.

Google translation produces the largest lexicons compared to the other translation methods. However, after manual filtering, the results retained are comparatively smaller. One of the reasons is that most of the entries are translated into phrases that are not multi-word expressions but rather e.g. clauses or clause portions.

Moses produces a small number of results since the training data were not be large enough and come from a different domain. Most of the entries from the English lexicons cannot be translated.

From the point of view of the source lexicon, one significant observation is that that SentiWordNet loses many of its entries in the filtering process. It is due to the entries from SentiWordNet that consist of a lot of specific names such as diseases, scientific names or terms, etc., that we consider as non-evaluative.

[10] `http://www.kamus.net`

Lexicon	Translation	Entries Obtained from Translation Pos	Neg	Entries after Manual Filtering Pos	Neg
Bing Liu	Google	1,147	2,589	740	1,500
General Inq		1,203	1,443	690	911
MPQA		1,429	2,426	796	1,359
SentiWordNet		3,404	4,857	873	1,205
Bing Liu	Moses	249	255	180	165
General Inq		379	245	237	130
MPQA		372	277	236	158
SentiWordNet		847	886	236	160
Bing Liu	Kamus	641	1,290	478	910
General Inq		884	1,009	536	692
MPQA		887	1,271	560	871
SentiWordNet		1,606	1,856	582	1,221

Table 4. Number of (de-duplicated) Indonesian expressions for each source lexicon and translation method before and after manual removal of wrong expressions

4.2. Merging Basic Lexicons

With the (many) baseline lexicons translated to Indonesian, we merged them by a)intersection and b) union. The basic idea of the intersection operation is to get the expressions that are agreed by different types of lexicons. The resulting lexicon should thus be smaller but with more validated expressions. On the other hand, the union operation is meant to greedily take all possible evaluative expressions. The intersection and union operations were performed on the lexicons from the same method of translation, lexicons with the same source of English lexicon, and also to all lexicons produced from translation. Table 5 shows the number of expressions for the lexicons from merging operations.

Looking at the intersection of lexicons from the same source, we can see that there is a significant drop in the number of negative expressions for SentiWordNet. We think that this is caused by the different translations provided by Google Translate and the online dictionary. The union operation, as expected, shows an increase in the number of expressions. The total number of unique entries after unioning all lexicons

[11] We exclude Moses-translated lexicons from the intersection with the source fixed and of the overall intersection because they contain too few entries.

Table 5. Positive and Negative Expressions after Intersection and Union

		Intersection[11]		Union	
Merging Lexicons of the Same…		Pos	Neg	Pos	Neg
Translation Method	Google	364	551	1256	1921
	Moses	92	78	366	246
	Online Dict	306	448	788	1565
Source	Bing Liu	330	660	932	1781
	General Inq	330	444	963	1185
	MPQA	376	619	1040	1638
	SentiWordNet	388	543	1112	1918
Merging All Lexicons Together		178	270	1557	2665

is significantly smaller than the union of their corresponding English lexicons, but still relevant considering the smaller number of expressions each lexicon has.[12]

4.3. Annotation-Based Lexicon

Since our annotation described in Section 3 include explicit markup of evaluative expressions in the sentences, we can extract a small lexicon directly from this data. In contrast to the general lexicons obtained above, this one is very much tailored to the examined domain.

5. Evaluation

We do not compare the lexicons directly to each other, but rather employ them in the practical task of predicting sentence polarity. We use a subset of our annotated sentences where the two annotators agree on the polarity as the test set. The test set consist of 380 sentences, 125 of which are labelled as positive, 13 as negative, and the remaining 242 as neutral.

5.1. Prediction Method

Given a lexicon a prediction method is needed to estimate the polarity of a given sentence. Our prediction method is very simple and identical for all the tested lexicons.

The polarity (sentiment) of a given sentence s that contains a set of positive expressions P and negative expressions N (with default weight 1.0) is predicted as:

[12]For comparison, the total number of expression of unioning all English lexicons is 7,809 for positive expression and 12,445 for negative one

$$\mathrm{polarity(s)} = \begin{cases} \mathrm{positive} & \sum_{p \in P} \mathrm{weight_{pos}(p)} > \sum_{n \in N} \mathrm{weight_{neg}(n)} \\ \mathrm{negative} & \sum_{p \in P} \mathrm{weight_{pos}(p)} < \sum_{n \in N} \mathrm{weight_{neg}(n)} \\ \mathrm{neutral} & \sum_{p \in P} \mathrm{weight_{pos}(p)} = \sum_{n \in N} \mathrm{weight_{neg}(n)} \end{cases} \quad (1)$$

When searching the sentences for positive and negative expressions (originating in the lexicon in question), we use the following constraints:

- **Unique Polarity.** An expression in a sentence can only be tagged with one type of polarity, either as positive or as negative expression.
- **Prioritize positive expressions.** If the lexicon lists the same expression both as positive and negative, ignore the negative one.
- **Prioritize longer expressions.** Since there was a possibility that a shorter expression is a part of a longer one, we collected the counts by first matching the longer expressions.
- **Negation.** We adapted technique presented in (Das and Chen, 2001) to handle the negation (inversion) of sentiment caused by a negation word. We used the words 'tidak', 'tak', 'tanpa', 'belum', and 'kurang' as negation words. The words that occur between the negation word and the first punctuation after the negation word were tagged with 'NOT_', e.g. 'kurang bagus gambarnya ?' (the picture is not good enough ?) to 'kurang NOT_bagus NOT_gambarnya ?'.

5.2. Performance Measures

We compare the performance of the lexicons using precision and recall of evaluative sentences:

$$\text{P-EVL} = \frac{c_{pos,pos} + c_{neg,neg} + c_{pos,neg} + c_{neg,pos}}{c_{pos,pos} + c_{neg,neg} + c_{pos,neg} + c_{neg,pos} + c_{neu,pos} + c_{neu,neg}} \quad (2)$$

$$\text{R-EVL} = \frac{c_{pos,pos} + c_{neg,neg} + c_{pos,neg} + c_{neg,pos}}{c_{pos,pos} + c_{pos,neg} + c_{pos,neu} + c_{neg,pos} + c_{neg,neg} + c_{neg,neu}} \quad (3)$$

where:

$c_{a,b}$: count of sentences with polarity a predicted as b
pos : positive
neg : negative
neu : neutral

Note that due to the small number of negative sentences in our test set, we examine the preformance only at the 'evaluative' level, i.e. we check how well the method distinguishes evaluative (positive or negative) sentences from neutral ones, disregarding the actual polarity.

5.3. Cross-Validation for Annotation-Based Lexicon

For a fair comparison, the lexicon extracted from our corpus of annotated sentences is evaluated in a 3-fold cross-validation.

The test data was randomly split into 3 sets (folds) of sentences. From each fold, we took the tagged positive and negative expressions, resulting in three lists of subjective expressions (lexicons). The prediction was then performed on each fold using the union of the lexicons taken from the other two folds. We report the average scores over the three folds.

5.4. Baseline and Oracle

In order to provide some context to our scores, we include the **Baseline** of predicting all sentences as evaluative. Marking all sentences as neutral gives P-EVL and R-EVL zero.

The **Oracle** performance is achieved if we extract the annotation-based lexicon from the complete test set and use it to predict the evaluativeness of the very same sentences.

5.5. Results

Figure 1 plots the precision and recall of all the lexicons.

The **Baseline** of marking everything as evaluative obviously has the recall of 100% and the precision is only 36%. The **Oracle**, as expected, had the highest precision (79%) and recall (90%) compared to the other type of lexicons.

The **Annotation-Based** lexicon, as cross-validated, maintains a very good precision (76%) but suffers a loss in recall, reaching only 54%.

The other observation that could be found was about the difference in performances of lexicons that were coming from different translation methods. Lexicons coming from Google translation had slightly higher precisions and recalls while Moses-translated lexicons performed worse esp. in recall.

As expected, intersecting lexicons leads to higher precisions at the expense of recall and unioning has the opposite effect. Google Translate again stands out here, bringing the highest recall when unioning across lexicon sources (u-G) and the highest precision when intersecting (i-G).

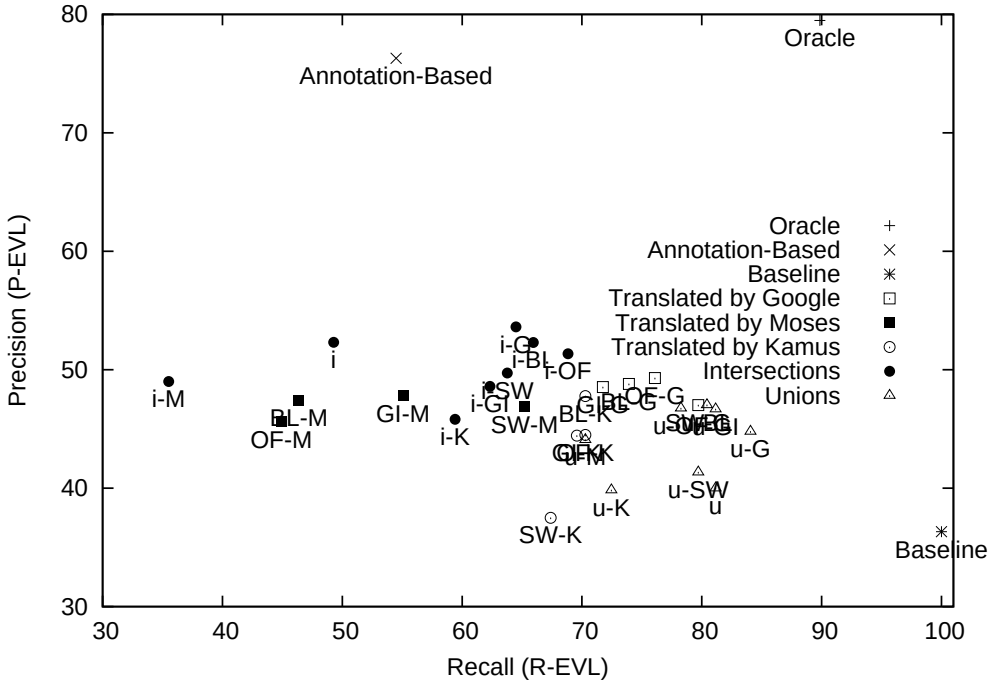

*Figure 1. Precision and recall of identifying evaluative sentences using various lexicons. The **Baseline** is to mark all sentences as evaluative.*

6. Weighting

The evaluations in Section 5 were done using lexicons that had expressions of weight 1.0. In realistic situation, the weight might vary, depending on how strong an expression projected the underlying positive or negative polarity.

We tried to assign this polarity strength to each expression. We used two different methods to achieve this objective. The two methods relied on the number of occurrences of the expression in a collection of 14,998 sentences coming from the same source of reviews but with no manual annotation.

The experiments in this section use the intersection of all Google-translated lexicons as the basis since this lexicon has a good balance of precision and recall.

- **Frequency Weighting**. The Frequency Weighting method assigned a weight that was the frequency or number of occurrences of the expression in the collection of the unannotated sentences. The basic premise was that the more often an expression is used in the review sentences, the higher its expressive value, assuming that the expression was used to express sentiment.

$$weight_{pos}(p) = freq_{all}(p) \quad (4)$$
$$weight_{neg}(n) = freq_{all}(n) \quad (5)$$

- **Iterative Weighting** In Iterative Weighting, an expression was given a weight of its relative frequency in the review sentences. For example, the weight a positive expression is equal to its frequency in positive sentences divided by its frequency in all of the sentences.

$$weight_{pos}(p) = freq_{pos}(p)/freq_{all}(p) \quad (6)$$
$$weight_{neg}(n) = freq_{neg}(n)/freq_{all}(n) \quad (7)$$

  Since the sentences used are unannotated sentences, we used the simple prediction method described in the previous section to first annotate the sentences. The default weight for each expression is set to 1.0. At the end of this annotation, weight of each expression is recalculated using the formula described. The prediction is repeated using these new weights, and so on until convergence.

6.1. Evaluating the Weighting Results

We showed the results of using the weighted lexicons to do prediction on test sentences in Figure 2. The accuracy and precision of lexicons with frequency and iterative weighting was lower compared to lexicon with default weight of 1.0. The significant difference was in the value of the recall. Putting weights on the expressions seemed

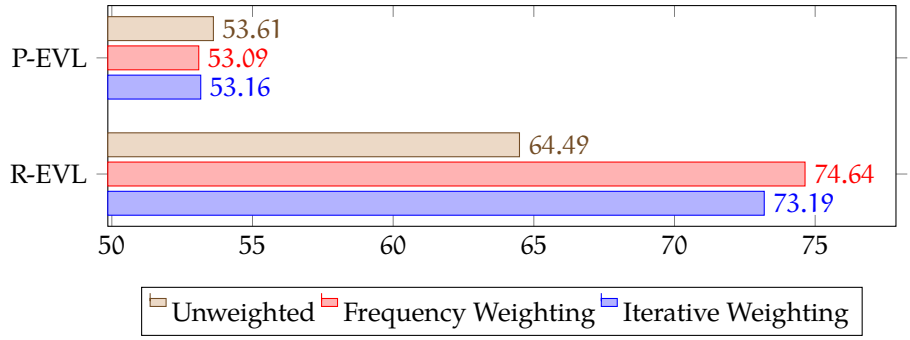

Figure 2. Impact of Frequency and Interative Weighting on evaluativeness prediction using intersection of Google-translated lexicons.

to be able to give a significant increase in the recall, with frequency weighting having higher recall than the iterative one.

The observed results confirm our expectations. Since we are evaluating only the evaluativeness of sentences and not their actual polarity, weighting has little effect on precision: sentences that contained expressions listed as positive in the lexicon will still contain them even if we reduce or increase their weight. The effect on recall can be attributed to sentences that contained an equal number of positive and negative expressions. Without weighting, the effect cancels out and the sentence is predicted as neutral. By introducing weights, we are very likely to break the balance and the sentence is predicted as evaluative one way or the other. The prediction thus marks more sentences and the growing recall confirms that these are correct sentences to mark – even humans labelled them as evaluative. If we were marking the wrong sentences, the recall would not increase and instead the precision would drop.

In Figure 3, we aim at increasing precision of the prediction. To this end, we remove expressions of low weight from the lexicon. Fewer sentences are thus going to be predicted as evaluative. Figure 3 plots the performance at the various thresholds. Only expressions with frequency higher than the threshold are included in the lexicon.

As we hoped for, excluding expressions of lower weight helps precision. However, the recall drops much faster than the precision grows.

7. Machine Learning Prediction

The simple prediction method described in Sections 5 and 6 does not consider any broader context as available in the input sentence. In general, we found several types of information that should be useful for the prediction:

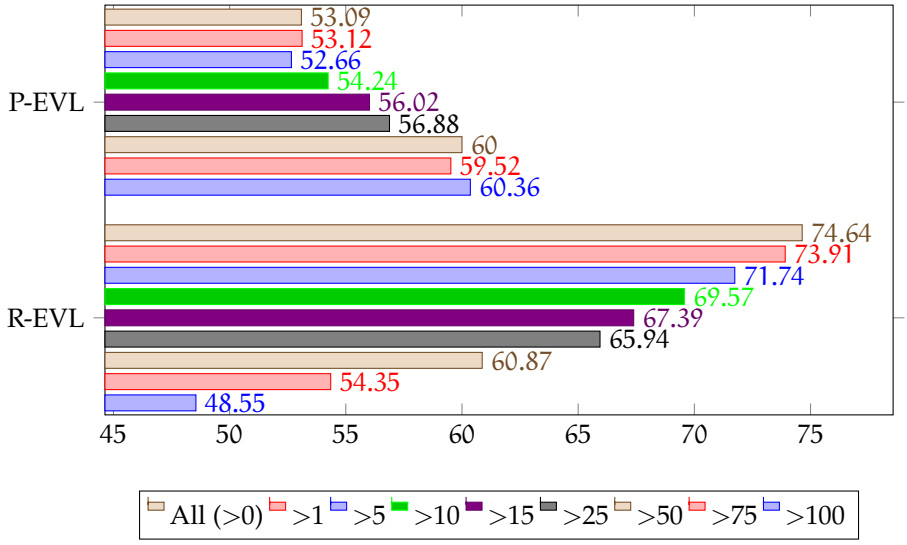

Figure 3. Thresholding using frequency weights.

- Overall Sentence Form
  A sentence with positive or negative expressions might not always be an evaluative sentence because the sentence structure or other features can prevent from such interpretation. We found several things that might cause the evaluative expressions to have no effect on the overall sentiment.
  The first case is when the sentence is in a hypothetical form, as in the example of 'sebuah keputusan yang salah akan membuat jiwa seluruh batalyon melayang percuma' (one wrong decision will cause the death of all batallions). In this sentence the word 'salah' (wrong) is identified as negative expression. However, this is only a hypothetical situation where the speaker expresses the opinion of what will happen, but not to evaluate the decision itself.
  Another structure that might affect the sentiment of the sentence is when it contrasts the positive and negative expressions as in the examples below:

  'menyuguhkan fitur yang berbeda, walau dengan model yang sama' (it comes with different features, though with the same design/model)
  'walau dengan model yang sama, menyuguhkan fitur yang berbeda' (though it has the same design/model, it comes with different features)

  In this context, the word 'berbeda' (different) is positive and 'sama' (same) is negative. Changing the parts of the sentence that are separated by a comma (one with 'though' and one without 'though') and depending on where the positive

and negative expressions are, the sentiment of the sentences can be different. The first sentence seems to be neutral and the second one seems to be more positive.
Questions are also mostly neutral, e.g., 'butuh ponsel yang murah tapi meriah?' (need a cheap and fancy phone?). The occurrence of evaluative expressions 'murah' (cheap) and 'meriah' (fancy) have no effect on the final sentiment of the sentence.

- Morphology and Multi-Word Expressions
  Some other information that can be useful is related to the word itself. The first such piece of information is the part-of-speech of the word. Some evaluative expressions might have a different meaning depending on what part-of-speech they take in a sentence. For example, the word 'menarik' can have meanings of 'pull' (verb), which can be considered as having no sentiment, and 'interesting' (adjective) which has positive sentiment.
  The other thing is that the evaluative expressions are sometimes used in a non-base form, e.g., 'indahnya' (how beautiful), which has the base form of 'indah' (beautiful). A simple word matching without lemmatization or stemming might not be able to capture the evaluative expression.
  Words that are part of larger phrases are also tricky and might cause an inappropriate detection of evaluative expressions, e.g., 'kurang lebih' (more or less), which contains the word 'kurang' (not enough) and 'lebih' (more/better). Predictions with simple word matching that we used in previous experiments are not able to capture this phrasal information.
- Target
  Information about the target of the discussion or target of the evaluation in an evaluative sentence is also important. Some sentences contain evaluative expressions that are not related to the main target of the discussion, e.g., 'selain bisa untuk berbelanja, website.com ... dengan foto-foto bayi anda yang lucu' (in addition to shopping, website.com ... with photos of your cute babies), where the target of the discussion is 'website.com' but contains a positive expression 'cute' for another target, the object 'baby'.

In this sections, we describe our experiment with machine learning techniques to include at least some of these ideas into the prediction method.

7.1. Features

Based on the previous observations, we defined a small set of 12 binary features which consisted of 10 non-lexicon related features (NonLexFeats) and 2 lexicon related features (LexFeats). The lexicon related features rely on one of the basic lexicons as used in the previous sections and they simply indicate whether at least one expression from the positive or the negative part of the lexicon was seen in the sentence.

The features are listed in Table 6.

Name	Type	Set to True if
Hypothetical	NonLex	Any of the words 'jika' (if), 'akan' (will), 'kalau' (if) appears in the sentence
Question	NonLex	'?' (question mark) appears in the sentence
Contrast 1	NonLex	Any of 'walaupun', 'meskipun', 'walau', 'meski' (though/although) is the first word of the sentence
Contrast 2	NonLex	Any of 'walaupun', 'meskipun', 'walau', 'meski' (though/although) appears anywhere except the first/last word
Negative List	NonLex	Any of the phrases 'cukup sampai disitu' (only until that point), 'kurang lebih' (more or less), 'salah satu' (one of the) appears in the sentence
Negation List	NonLex	Any of the words 'tidak' (not), 'tak' (not), 'tanpa' (without), 'belum' (not yet), 'kurang' (less), 'bukan' (is not) appears in the sentence
Adjective Word	NonLex	Any adjective (surface) words appears in the sentence
Adjective Lemma	NonLex	Any adjective lemmas appears in the sentence
Question Word	NonLex	Any question (surface) words, e.g. 'apakah' (what), 'bagaimanakah' (how), appears in the sentence
Question Lemma	NonLex	Any question lemmas, e.g. 'apa' (what), 'bagaimana' (how), appears in the sentence
PosLex	Lex	At least one of the positive expressions from the lexicon appears in the sentence
NegLex	Lex	At least one of the negative expressions from the lexicon appears in the sentence

Table 6. Features for sentiment prediction

7.2. Evaluation Setup

For the experiment, we used *scikit-learn* [13], a machine learning library for Python. We chose SVM as the machine learning method and used the default function *svm.SVC()* provided by the library. The kernel used by this function is an RBF kernel, and we just used the function with its default parameters.

The evaluation was performed using 3-fold cross validation with the same division as in Section 5. The results shown here are the average value across the three runs.

7.3. Comparing the Features

We compared the performances of using LexFeats only (lexicons only), using Non-LexFeats features only, and using all of the features (AllFeats). Figure 4 compares the average performances of these three setups. The averaging of LexFeats and AllFeats

[13] `http://scikit-learn.org`

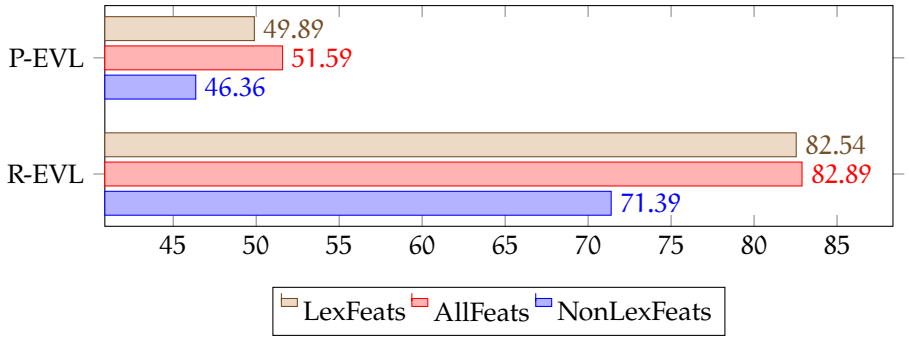

Figure 4. Average performances using various feature categories

were done across all different types of lexicons and of the three folds of test data used. The average of NonLexFeats were done only on the three folds of test data, since they were not using any lexicons in the prediction.

The results indicate that using additional features other than the lexicons improves both precision and recall, although with a rather small margin. Using only the two features of LexFeats seems to produce better results than just the NonLexFeats.

7.4. Comparison with the Simple Prediction Method

We would like to see how the machine learning prediction performance compares to the performance of the simple prediction. In order to objectively compare these two different predictions, we use the very same 3 folds for both methods and plot averaged precisions and recalls, see Figure 5.

The obvious difference that we observed was the performances of the recalls were increasing in machine learning prediction. All lexicons seemed to have high recalls, compared to the simple prediction method that had more scattered recall values. The precisions, however, showed no improvements and stayed below 60%.

8. Conclusion

We introduced two resources for Indonesian sentiment analysis: 446 annotated sentences and a collection of subjectivity lexicons constructed by manually filtering the results of automatic translation of subjectivity lexicons available for English.

The annotation of the review sentences shows the nature of the data: it mostly consists of neutral sentences. The evaluative sentences are primarily positive, so there is just a handful of negative sentences in our dataset. The inter-rater agreement (κ) for deciding whether a sentence is evaluative or not is 0.697. However, the agreement

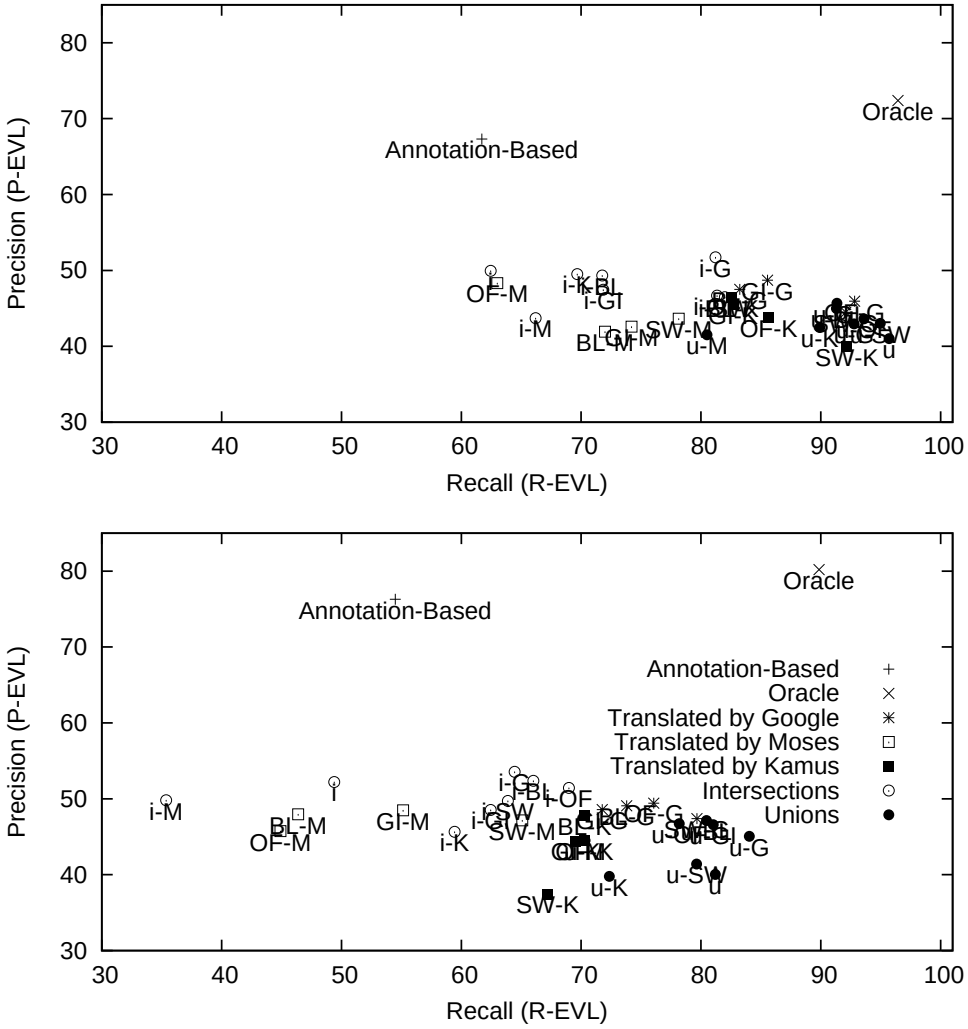

Figure 5. Precision-recall graph using our machine-learning setup (upper chart) and using the simple prediction (lower chart) in identical 3-fold cross validation.

on the actual polarity for the 140 evaluative sentences (where both annotators marked the sentence as evaluative) is surprisingly high, reaching κ of 0.921.

We produced 12 basic lexicons built by automatic translation and 16 lexicons by intersecting and unioning. The average number of expressions is 1,285 for the basic

lexicons, 747 for lexicons from intersection operations and 2,617 for lexicons from union operations.

The combination of different sources of lexicons, translation methods, and merging operations gives rise to lexicons with different numbers of entries that share some evaluative expressions but also possess their own unique expressions.

Evaluations performed on the resulting lexicons using simple prediction method show that the lexicon from intersection of Google translation of all source lexicons results in the highest precision. In terms of recall, the union of Google translations gets the highest score. The other interesting result is that a very small baseline lexicon extracted directly from (a heldout portion of) the training data achieves much higher precision than all other lexicons.

The weighting experiments that we have conducted show that the weights might help in increasing recall, although the trade-off of losing the precision exists.

We also tried to replace the basic prediction method with machine learning. This allows to incorporate other helpful information, not related to the lexicons. This helps to increase the recall for all types of the lexicons but there is no improvement in precision.

Bibliography

Baccianella, S., A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta, May. European Language Resources Association (ELRA)*, 2010.

Bakliwal, Akshat, Piyush Arora, and Vasudeva Varma. Hindi subjective lexicon: A lexical resource for hindi adjective polarity classification. In Chair), Nicoletta Calzolari (Conference, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

Banea, C., R. Mihalcea, and J. Wiebe. A bootstrapping method for building subjectivity lexicons for languages with scarce resources. In *Proceedings of LREC*, 2008.

Das, Sanjiv and Mike Chen. Yahoo! for amazon: Extracting market sentiment from stock message boards. In *Proceedings of the Asia Pacific Finance Association Annual Conference (APFA)*, volume 35, page 43, 2001.

Gabrielatos, Costas and Anna Marchi. Keyness: Matching metrics to definitions. *Theoretical-methodological challenges in corpus approaches to discourse studies-and some ways of addressing them*, 2011.

Hu, M. and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM, 2004.

Kaji, Nobuhiro and Masaru Kitsuregawa. Building lexicon for sentiment analysis from massive collection of html documents. In *Proceedings of the joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 1075–1083, 2007.

Maks, Isa and Piek Vossen. Building a fine-grained subjectivity lexicon from a web corpus. In Chair), Nicoletta Calzolari (Conference, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

Miller, George A. Wordnet: a lexical database for English. *Communications of the ACM*, 38(11): 39–41, 1995.

Pérez-Rosas, V., C. Banea, and R. Mihalcea. Learning sentiment lexicons in spanish. In *Proc. of the 8th International Conference on Language Resources and Evaluation (LREC'12)*, 2012.

Smedt, T.D. and W. Daelemans. "vreselijk mooi!" (terribly beautiful): A subjectivity lexicon for Dutch adjectives. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC12)*, 2012.

Wilson, T., J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT/EMNLP 2005*, 2005.

Address for correspondence:
Ondřej Bojar
bojar@ufal.mff.cuni.cz
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics,
Charles University in Prague
Malostranské náměstí 25
118 00 Praha 1, Czech Republic