Extracting Contextonyms from Twitter for Stance Detection

Guillaume Gadek[1,2], Josefin Betsholtz[1], Alexandre Pauchet[2], Stéphan Brunessaux[1], Nicolas Malandain[2] and Laurent Vercouter[2]

[1]*Airbus DS, 78990 Elancourt, France*

[2]*Normandie Univ, INSA Rouen, LITIS, 76000 Rouen, France*

Keywords: Opinion Mining, Context, Contextonyms, Sentiment Analysis, Social Media Data, User Generated Text.

Abstract: Opinion mining on tweets is a challenge: short texts, implicit topics, inventive spellings and new vocabulary are the rule. We aim at efficiently determining the stance of tweets towards a given target. We propose a method using the concept of contextonyms and contextosets in order to disambiguate implicit content and improve a given stance classifier. Contextonymy is extracted from a word co-occurrence graph, and allows to grasp the sense of a word according to its surrounding words. We evaluate our method on a freely available annotated tweet corpus, used to benchmark stance detection on tweets during SemEval2016.

1 INTRODUCTION

The large volume, easy access and rapid propagation of online information make the Internet a perfect medium for opinion mining. In particular, Twitter has emerged as a micro-blogging community with over 500 million public messages per day.

Similar to SMS, tweets are short, contain inventive spelling and their meanings are often implicit. However, they also differ (Gotti et al., 2013): tweets can be public whereas SMS are strictly private. Twitter also contains a greater extent of invented words and typing errors (Maynard et al., 2012). Furthermore, tweets contain entities such as hashtags[1] or user mentions[2]. Both hashtags and user mentions often appear as labels, without any syntactic role.

Nevertheless, most algorithms adapted to tweets consider words as atoms, i.e. without considering any relation to the surrounding words, which generates ambiguity as most words can have more than one meaning. The problem of discovering the "real" sense of words in a text is commonly referred to as *text disambiguation*. We propose to use *contextonyms* to solve this issue, following the simple idea that the surrounding words can be exploited to determine the context of usage of a word, and therefore to determine its sense. Two words are contextonyms if they are commonly used in a same context and a group of words frequently co-occurring is called a contextoset.

Twitter is commonly used to express views on various topics, including product reviews and political opinions. In this domain, an opinion is represented by five elements: its author, time of utterance, target (e.g. a phone), aspect of the target (e.g. the *screen* of the phone), and polarity/sentiment (Pang and Lee, 2008).

Stance detection is a similar idea: it focuses on determining the polarity (in favor, against or none) of an opinion. Texts are gathered by *topic* and have to be analyzed with regard to a given *target*. To clarify the difference, if the target is Hillary Clinton, the topic could be U.S. Election Candidates or U.S. Female Politicians.

This task[3] requires an excellent knowledge of the topic as implicit statements have to be identified: the corpus includes utterances towards different entities, related to the topic but sometimes different from the target. As an example, among the SemEval task was a "Hillary Clinton" corpus, which included tweets targeting Donald Trump, her rival in the 2016 US election.

In this article, we propose to use contextonymy, and more precisely contextosets, to disambiguate tweets. We outline a new method to extract the contextosets and show that this process improves the

[1]Hashtags are single words or phrases preceded by a # and whose meaning may be maintained in the sentence. For instance, "#voteforyou" can replace "vote for you".

[2]User mentions use *@name* in place of a named entity.

[3]SemEval proposed a stance detection task on tweets for the first time in 2015 (linked to the 2016 NAACL conference): http://alt.qcri.org/semeval2016/task6/.

Gadek G., Betsholtz J., Pauchet A., Brunessaux S., Malandain N. and Vercouter L.
Extracting Contextonyms from Twitter for Stance Detection.
DOI: 10.5220/0006190901320141
In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017)*, pages 132-141
ISBN: 978-989-758-220-2

stance detection on tweets. To benchmark our approach, we compare our results on the SemEval task of stance detection on tweets.

Section 2 introduces others' work on stance detection, word sense disambiguation, and contextonyms. Then, Section 3 explains how we extract contextosets and determine stance. Section 4 presents the implementation and experiments. Finally, Section 5 shows the results obtained by our method and highlights some elements to discuss.

2 RELATED WORK

Previous studies on opinion mining and sentiment analysis have already proposed excellent methods, but only a few of them are applicable in stance detection.

2.1 From Opinion Mining to Stance Detection

Stance detection techniques can be divided into three main approaches: first, the use of sentiment dictionaries and specific linguistic rules. Second, machine learning for text categorization, using a training corpus. Third, hybrid methods that combine the two first approaches. The three techniques are described in the following sections.

Resource-based Approaches for Sentiment Analysis (SA). This family of methods associates polarities with words, i.e. each word gets a score reflecting its degree of "positivity" and a score reflecting its degree of "negativity". A weighted sum, usually called valence, produces the overall sentiment of a text.

Vader[4] (Hutto and Gilbert, 2014) is a rule-based sentiment analyzer. Inspired by LIWC[5], ANEW[6] (Bradley and Lang, 1999) and SentiWordNet[7] (Baccianella et al., 2010), the authors compiled a short list of 7,500 word-valence pairs, and benchmarked it successfully.

Some systems quite accurately predict sentiment, but this indicator is not satisfactory for stance detection in politics. (Tsytsarau and Palpanas, 2012) proposes to detect contradiction: they focus on the signs of opposition or agreement between two successive posts. A study on UK politics (Maynard and Funk, 2011) uses GATE (Cunningham et al., 2011), a rule-based tool, and applies it to political opinion detection. A set of rules states whether a sentence is for or against one of the three big parties in the UK. Similar work focuses on online debates (Anand et al., 2011), to distinguish messages in terms of opposition to entities: many posts are exclusively negative, and therefore a "positive/negative" vocabulary approach would not be useful.

Learning-based Approaches. A second way, Text Classification, relies on a human annotated training corpus. This approach stems from topic categorization (Pang and Lee, 2008), which objective is to label a document with predefined categories. The impact of the chosen features is important, as (Tan et al., 2002) showed by comparing unigrams and/or bigrams of words.

Social, user-generated contents are very specific to a given platform. The increase of inventive spellings on the Internet led (Pak and Paroubek, 2010) to gather a corpus for sentiment analysis, classifying tweets as objective (thus, neutral) or subjective (and then, relying on laughing and crying emoticons, between positive and negative). In 2013, on a contextual polarity task, best results used SentiWordNet or MPQA[8] (Wilson et al., 2005) and achieved a F_1-measure of 0.68 (Nakov et al., 2013): the task is difficult and no good system has emerged yet.

Hybrid Approaches. (Andreevskaia and Bergler, 2008) combined the two precedent approaches: lexicon-based and learning-based. In their experiments, they proposed a corpus consisting of debate forum posts, such as "Firefox VS Internet Explorer". A post in a forum thread is usually part of a discussion, which helps to put the elements in relation.

(Hasan and Ng, 2013) reached a 0.75 accuracy on political forum posts stance classification. They used multiple approaches: the presence of word unigrams and bigrams, sentiment features using the LIWC resource (Pennebaker et al., 2001), as well as task dependent features: they assume that an author keeps the same stance throughout a debate (ideological constraint), and that two successive messages have a great probability to oppose each other (user-interaction constraint). These types of features are not applicable in the SemEval corpus, as it only contains the text of tweets.

[4] Valence Aware Dictionary for sEntiment Reasoning.

[5] Linguistic Inquiry and Word Count: in a text, it counts the percentage of words that reflect different emotions, thinking styles, social concerns, and parts of speech.

[6] Affective Norms for English Words: a set of normative emotional ratings for a large number of words in the English language.

[7] A lexical resource for opinion mining. It assigns sentiment scores to synsets from WordNet.

[8] Multi-Perspective Question Answering: this corpus contains news articles from a wide variety of news sources, manually annotated for opinions and other private states.

To benefit from the strengths of sentiment resources, (Khan et al., 2014) combined three methods: an emoticon classifier, an enhanced emotion classifier and a SentiWordNet classifier. Results show improvements in comparison to the methods taken one by one.

2.2 From Words to Relations

The previous section briefly reviewed the tools to detect the stance of a text. We believe that the scores can be improved if we tackle one of the biggest challenges: the ambiguity of the texts. In the following part, we review some techniques for word sense disambiguation.

Word sense disambiguation consists in choosing between senses when the meaning of a word is not obvious. Most words can bear more than one meaning and their senses can be detected from the context in which the words appear. (Wiebe and Mihalcea, 2006) show an improvement of their sentence classifier when considering the feature "subjectivity information" (syntactic rules to determine if the sentence is subjective or objective).

The exploration of various links between words is an active field: (Rei and Briscoe, 2014) look for hyponyms in a word embedding space. Hyponyms generation allows more relations per word, as opposed to synonyms or antonyms. (Perez-Tellez et al., 2010) attack the homonymy problem[9]. They claim to be able to distinguish between "orange", the color, and "orange", the fruit. To achieve this, they convert each tweet to a tf.idf[10] vector and then apply clustering (K-means) on the whole corpus: tweets in a same cluster tend to use the words in the same sense.

(Fernando and Stevenson, 2012) aimed to associate a Wikipedia article to each of the WordNet synsets: first by matching the title to generate a candidate list, then refining this selection by considering the whole article as well as the glossary and description fields of the synsets.

Previous work (Zesch et al., 2008) proposed a semantic relatedness score using concept vectors to map documents, comparing various resources such as Wiktionary, Wikipedia articles, WordNet and GermaNet (a german avatar of WordNet).

More recently, (Feng et al., 2015) claim that the usual methods for semantic relatedness of words, using WordNet (Miller, 1995) or Wikipedia, give poor results on Twitter content, because of the different sentence structure as well as the presence of new vocabulary.

These techniques have been used on many different types of corpus, but our work focuses on social media data. User-generated texts are very different from other corpora, however, contextonyms may help in our stance detection task.

2.3 Contextonyms and Contextosets

The concept of *Contextonyms* was first introduced by (Hyungsuk et al., 2003), noting that "contextually related words are meaningful indicators of a target word's semantic value in a given context". In this study, contextonyms are defined as "relevant contextually related words for a target word". In turn, "context" is defined as a certain number of surrounding words. Contextonymy is a relation between words, as is synonymy: two words are contextonyms if they frequently occur together (and thus, describe the same context).

(Hyungsuk et al., 2003) also obtained *cliques* - complete subgraphs - from the contextonyms, which, according to them, represent the minimal senses of words. We estimate that the *target word* used by (Hyungsuk et al., 2003) is not enough to characterize a context. Still, a contextoset regroups a number of words corresponding to a given context.

Works using Contextonyms. (Ploux and Ji, 2003) and (Wang et al., 2016) propose a Statistical Machine Translation method, where the unit is not a word but a "minimal semantic unit", represented by a clique (following (Hyungsuk et al., 2003) contextonyms extraction). (Şerban, 2013) extracted the contextonyms from movie subtitles, to correct SentiWordNet by continuing the propagation of sentiment valences along these relations.

Comparison of Contextosets, Word Embeddings and Synsets. To assess the semantic relatedness proposed by *contextosets*, we compared the results with the outcome of two other methods: Word2Vec and WordNet. We used the same 70 million word corpus, completely composed by tweets, in all methods.

We obtained word embeddings by training a Word2Vec model (Mikolov et al., 2013) using "bag-of-words" and a vector size of 100. We then extracted contextonyms, using the procedure outlined in this paper. Finally, focusing on the word "support", we acquired the synsets from WordNet. Table 1

[9]When two words that have different meanings are either spelled in the same way (e.g. "match" (that you light a fire with) and "match" (a sports game) or pronounced in a similar way (e.g. "to" and "too").

[10]Term frequency . inverse document frequency: a numerical statistic reflecting how important a word is to a document in a corpus.

shows an excerpt of neighboring words, contexosets and synsets for the word "support".

From the Word2Vec word embeddings, only some words seem related to our target, and the results largely depend on the training corpora. The grammatical categories are not considered important, and we have no insight into the relations between the other words ("respect" is not included in the set of the closest words to "organize", whereas they both appear as close words to "support").

WordNet synsets are sometimes numerous, but do not exist for Twitter-specific vocabulary. Even the very popular "LOL" is excluded from this dictionary: WordNet is legitimate, but it is not adapted for social media.

Contextonyms, like Word2Vec, make no distinction between grammatical categories and may include any word that has been used in the corpus. Furthermore, they quite efficiently convey the meaning of a word, and the contextosets evidently match the original corpus topics.

2.4 Discussion

During the SemEval2013 opinion mining task, the best system achieved a F_1-measure of 0.68 (Nakov et al., 2013). (Andreevskaia and Bergler, 2008) are a reference in stance detection: on the topic "Firefox VS Internet Explorer", their hybrid classifier achieved $F_1 = 0.66$. On another dataset, (Hasan and Ng, 2013) reached a 0.75 accuracy. These results are good, but not excellent: the stance detection task is indeed very difficult. Moreover, during annotation, the inter-annotator agreement[11] is often quite low, which means that even if the stance detection procedure performs well on the testing data, it is still debatable whether the results truly represent the actual stances.

We believe that stance detection on tweets process could benefit from disambiguation techniques. However, due to the difficulties of social media contents, we cannot rely only on well-established resources. Disambiguation itself also presents great challenges: (Perez-Tellez et al., 2010) reached a F_1-measure of 0.74 on some ambiguous company names. Nevertheless, we believe that word sense disambiguation has great potential and including such a step in stance detection can have a positive impact. Contextonymy appears to have many advantages that makes it suitable to use for disambiguation. Hence, the aim of this study is to develop a method to extract a database of contextosets from tweets, and then exploit this database to disambiguate the senses of tweets in order to improve stance detection.

[11] On an annotation task, each sample is labeled by different persons to check if they agree; measures such as Kohen's kappa enables evaluation of their agreement.

3 EXTRACTION AND USE OF CONTEXTOSETS

3.1 Contextoset Extraction

Required Resources. Contextosets are extracted from a corpus of documents. The contents of these documents should be representative of the topics of interest. Here we consider a corpus constituted of tweets only. In order to obtain as meaningful contextosets as possible, a thematic corpus should be used.

Preprocessing. In this step, all tweets are lowercased, and user mentions, special symbols, and stop words are removed. Common abbreviations are converted to their full words (e.g. "I'm" becomes "I am"). The tweets are then tokenized on their white spaces. We considered using a lemmatizer and POS-tagger, however, our tests with TweetNLP (Owoputi et al., 2013) were not convincing, as the model lacks information about the specific topics in our corpus. For instance, the word *ISIS* was "lemmatized" to *IS*, which completely eliminates the meaning of the word.

Constructing a Co-occurence Graph.

Definition 3.1. *Tweet.* A tweet t is a set of words $\{n_i, n_j, ...\}$ obtained from preprocessing a real tweet.

Definition 3.2. *Co-occurrence.* Words n_1, n_2 are said to occur together if they are in the same tweet t and they are separated by less than $WindowSize/2 - 1$ words. Stated differently, a word co-occurs with the $WindowSize/2$ words before it and the $WindowSize/2$ words after it, in the same tweet t. $WindowSize$ is a parameter that can be set to any even, positive integer.

Using a corpus of preprocessed tweets as in Definition 3.1, we constructed a co-occurrence graph $G = (V, E)$. The set of nodes $\{V\}$ consists of the complete vocabulary of the preprocessed corpus, and the set of edges $\{E\}$ represents the undirected, valued links between all pairs of co-occurring words (see Definition 3.2). The weight w_e of any individual edge e is the number of co-occurrences of the words linked by e in the entire preprocessed corpus.

Table 1: Word Embeddings, contextosets and WordNet synsets for the nearest words of **support**.

Method: Word Embeddings
supporting, supported, supports, respect, vote, encourage, voting, voted, organize, helping
Method: Contextosets
(support, continued, foolery), (climate, support, advocacy, preventing, change), (support, bae, naten, kanta), (support, tennessee, thank, trump2016)
Method: Synsets
(documentation, support) (support, keep, livelihood, living, bread and butter, sustenance) (support, supporting) (accompaniment, musical accompaniment, backup, support) (support, financial support, funding, backing, financial backing) (support, back up) (back, endorse, indorse, plump for, plunk for, support) (hold, support, sustain, hold up) (confirm, corroborate, sustain,substantiate, support, affirm) (subscribe, support) (corroborate, underpin, bear out, support) (defend, support, fend for) (patronize, patronise, patronage, support, keep going) (digest, endure, stick out, stomach, bear, stand, tolerate, support, brook, abide, suffer, put up)

Filtering Words. As previously mentioned, one of the great challenges of interpreting tweets is that they do not necessarily comply with established rules of grammar and spelling. Moreover, many new words have emerged specifically on social media, and they often convey important clues about the content and/or stance of the tweet. For instance, the hashtag "#demexit" implies US democrats leaving the democratic party, and has been frequently used in discussions about the 2016 U.S. election on Twitter. However, this word, like many others, is absent in the Oxford Dictionary. Other words, such as "laaazzzyymoonnddayyy" do not belong to "established" Twitter-vocabulary, but are purposely misspelled words used by one person alone. It is important to be able to separate the important social vocabulary and the nonsense words when creating contextonyms. Furthermore, if a tweet containing a nonsense word is retweeted many times this word will appear important by conventional filtering methods, which remove low-frequency words, relatively to an actual word which perhaps only occurs once or a few times in the corpus. Therefore, we have developed an innovative method to filter out the non-usable words.

Definition 3.3. *Degree.* The degree of a word n in a co-occurrence graph G is the number of other words directly connected to n. We denote this by $d(n)_G$.

We determine a word to be legit if it is used in many different kinds of contexts, i.e. surrounded by different words. This simply indicates that the word is present in more than one tweet and/or is used by more than one person. One way to assess this is to look at the degree of a node. However, since we rely on *WindowSize* parameter to assign neighbors to a word, its degree is also dependent on its position in any given tweet. For instance, "dogs like to swim in the summers" and "dogs usually run very fast" would give "dogs", "swim", "to", "in", "run" degrees of 4, even if "dogs" is the only word that appear in both sentences. Therefore, we normalize the degree of a node by its average degree due to its position in a tweet to get a ratio, α, that represents the actual variety of contexts that a word appears in.

Let $g_t = (V_t, E_t)$ be the co-occurrence graph for a single tweet t. For a given word n, let the tokenized tweets containing n be denoted by 1,...,K. Then, the average degree ϕ of a word n, due to its position, is given by

$$\phi(n) = \frac{1}{K} \sum_{j=1}^{K} d(n)_{g_j} \tag{1}$$

We can then find $\alpha(n)$, the ratio of degree in G to average degree position for word n, to be

$$\alpha(n) = \frac{d(n)_G}{\phi(n)} \tag{2}$$

A large score implies that word n occurs in a great variety of contexts. Hence, the words in a tweet such as "dizz movi ezz hoorrble", if only appearing in this

tweet, would all only get a score of 1, even if the tweet is retweeted 50 times. A word n would then be removed if $\alpha(n) < \alpha_{threshold}$.

The second part of the filtering process concerns the edges. Again, the conventional filtering method, that filters by edge weight, would remove important contexts that are less represented in the corpus, perhaps because of topic bias, and favor co-occurrences that appear frequently, even if it is by retweet. In an attempt to address this issue, we introduce metric β, that consists of two weight-node count ratios.

$$\beta(e) = \frac{w_e}{c_{n_{1,e}}} + \frac{w_e}{c_{n_{2,e}}} \quad (3)$$

Where w_e is the weight of edge $e = (n_1, n_2)$, $c_{n_{1,e}}$ and $c_{n_{2,e}}$ are the word counts for the two words n_1 and n_2 connected by e. Since $\beta_e =]0,2]$, a value approaching 2 implies that this association is very important for both words, whereas a value approaching 0 implies that the association is relatively unimportant for both words. By filtering away the edges that have small values, i.e. whenever $\beta_e < \beta_{threshold}$, we get rid of the unimportant associations and only retain what is important for our contextosets.

Contextoset Extraction. We chose to extract the contextosets using a method proposed by (Palla et al., 2005). They outline a way to obtain *k-cliques*, that is, communities derived from overlapping cliques. This seems to suit our problem particularly well as maximal cliques provide inadequate contextosets because of the tendency to form many, almost identical sets of words that specify the same context. *k-cliques* improve the contextosets as they merge cliques that share many of the same words. We used the *k_clique_communities* implementation of (Palla et al., 2005) method in the NetworkX (Hagberg et al., 2008) python package.

3.2 Determining Stance

As described in Section 2, there are two main approaches for stance detection. The first is based on sentiment analysis, guided by the intuition that positive-sentiment tweets have a supportive stance towards their target. The second one is based on text categorization, where conditional probabilities of word co-occurrences help to statistically determine the class to which a tweet belongs. For each of these approaches, we propose a baseline and a method using contextosets. We do not claim to have the best classifier, but we aim to show that contextosets can substantially improve stance detection.

3.2.1 Sentiment: Resource-based Approaches

Baseline, SENT-BASE. We propose a baseline, SENT-BASE, using the well-known resource SentiWordNet 3.0 (Baccianella et al., 2010) to predict the stance. We assume that positively (negatively) valued tweets have the stance *FAVOR* (*AGAINST*).

In SentiWordNet, each word n may be present in different *synsets*. A synset is a set of one or more synonyms that are interchangeable in a given context. Let $S(n)$ be the set of i synsets s_i containing the word n. Each synset has a positive and a negative valence s_i^+, s_i^-.

Let S_t be the set of all the N synsets taken into account for the whole tweet. We therefore define the valence $v(t)$:

$$v(t) = \frac{1}{N} \sum_{s_i \in S_t} s_i^+ + s_i^- \quad (4)$$

If $v(t)$ is positive (negative), we assume the tweet is supportive (opposed), thus having a stance *FAVOR* (*AGAINST*).

Enhancing Sentiment Analysis with Contextosets: SENT-CTXT. Sentiment prediction can be improved by considering contextonyms when selecting synsets. We obtain a list of the best contextoset(s) matching a tweet, by criterion of the greatest number of words shared by the contextoset(s) and the tweet. It is possible to have more than one contextoset sharing the same number of words with the tweet.

Let C be the set of contextosets c generated from the corpus. Then, for a tweet t, the set of best contextosets B_t is given by

$$B_t = \{c \mid max(|\{n\}_c \cap \{n\}_t|), \forall\, c \subset C\} \quad (5)$$

Then, we propose a function that takes the tokenized tweet and its contextoset(s) as inputs. Using SentiWordNet, it selects only one synset based on the shared number of words between the contextonyms and the synset. If two synsets are competing, the function relies on the "gloss" field (glossary, a short unstructured description) to count the number of shared words and, finally, select the best synset.

Finally, the valence is computed as in SWN-BASE and allows us to predict a stance label.

3.2.2 Statistical Approaches

Baseline: SVM-UNIG. We propose a baseline, SVM-UNIG, using a SVM on word unigrams. More specifically, we compared different algorithms and parameter settings and finally selected and trained a

SVM with RBF kernel ($C = 100.0$, $\gamma = 0.01$ after a 3-stratified folded cross validation). The feature vector is composed of the boolean indicators of the unigrams presence. Vocabulary size is fixed at 10,000, which limits the feature vector length.

Using Contextosets as Features: SVM-CTXTS. SVM-CTXTS is following a simple intuition: perhaps the contextosets are good indicators of the stance of a tweet. Thus, SVM-CTXTS is the same classifier as SVM-UNIG, but the feature vector is here composed of the boolean indicators of the presence of a contextoset. However, we believe this method is sensitive to the size of the training set: due to the limited size of our training and test sets (1250 tweets in the test set), and because we obtained 6278 contextosets, it is likely to see contextosets occur only once in a while.

Using Contextosets to Expand Tweets: SVM-EXP. This method address to the shortness of the tweets by completing them with the best contextoset(s). It first transforms the tokenized tweet by adding all words of the best matching contextoset(s). Then, it uses SVM-UNIG to determine its stance. Of course, this process has to be applied both on the training set and on the testing set.

We use Equation 5 to find the best contextosets B_t associated to the tweet t, and then obtain the "expanded tweet" E_t as follows.

$$E_t = \{n\}_{B_t} \cup \{n\}_t \tag{6}$$

3.3 Evaluation

For each possible label s, the classifier can send a *positive* or a *negative* signal in response to a query sample. The response signal can be either *true* if it matches the ground truth, or *false* in case of error.

To compute metrics to evaluate our results, a set of samples called the *test set* is needed.

Thus, we have four possible outcomes for each label s: *TP* stands for the number of true positives, meaning that the classifier correctly determined TP samples to belong to s; *FP* stands for the number of false positives, meaning that the classifier incorrectly determined the label of FP samples to be s; *TN* represents the number of true negatives, meaning that the classifier correctly determined *TN* samples not to belong to s; and *FN* stands for the number of false negatives, meaning that the classifier incorrectly determined *FN* samples not to belong to s.

The first metric, *Precision*, is defined in Equation 7 and represents the fraction of *true positives*, given samples (whose true labels are of mixed categories) that were all classified to belong to s. Another metric, *Recall* (Equation 8), determines the fraction of *true positives*, given samples (whose true labels are all s) that were classified to belong to any category. Together, we combine them to obtain the F_1-measure (Equation 9) which is commonly used to assess the quality of the prediction for each label s.

$$P_s = \frac{TP_s}{TP_s + FP_s} \tag{7}$$

$$R_s = \frac{TP_s}{TP_s + FN_s} \tag{8}$$

$$F_1(s) = 2\frac{P_s R_s}{P_s + R_s} \tag{9}$$

To benchmark our results, we use the metric *Official Score* proposed in the SemEval task (Equation 10). It consists of the average between the positive (F for favor) and negative (A for against) stances F_1-measures (and does not include the neutral-stance prediction). It is not directly comparable to F_1-measures given beforehand.

$$Score = \frac{1}{2}\ (F_1(F) + F_1(A)) \tag{10}$$

4 IMPLEMENTATION AND EXPERIMENTS

4.1 SemEval Task Description

Our stance classifiers are evaluated on the SemEval2016-task6 corpus[12]. We focus only on the subtask-A, which includes a training set and a test set on five topics ("Atheism", "Climate Change is a Real Concern", "Feminist Movement", "Hillary Clinton", "Legalization of Abortion"). Subtask-B focused on "Donald Trump" in an unsupervised way, with only a test set available. Overall, the subtask-A corpus is divided in a training part (2,914 tweets) which is used to find best parameters and train our supervised model, and a test part (1,250 tweets) which is used for evaluation. To compare the results, they use the *Official Score* (see Equation 10).

4.2 Input Corpora for Contextosets Extraction

We collected a corpus of English-written tweets, *GenTweets*, using the Twitter Stream API. This API allows anyone to gather public tweets that contain one

[12]SemEval2016-task6 is a stance detection task applied on an annotated tweet corpus freely available at http://alt.qcri.org/semeval2016/task6/

or more keywords (words or hashtags), as soon as they are published. However, due to constraints imposed by Twitter, we only receive a random sample of the set of tweets containing these keywords. *GenTweets* consists of 7,773,089 tweets gathered between November 20th and December 1st, 2015, on a broad range of topics, including Clinton, the abortion debate, religion, and miscellaneous.

4.3 Parameters

Using the *GenTweets* corpus, we obtained a vocabulary size larger than 250,000 words: we restrained it at 50,000 setting $\alpha_{threshold} = 10$. This size is more reasonable to handle and allows for easier processing of the co-occurrence graph, yet includes enough variety to grasp a wide range of expressions.

We chose $\beta_{threshold} = 0.06$ as this this number implies that the link between two words is of relatively small importance; it also limits the number of edges at 300,000, again allowing for reasonable processing times.

5 RESULTS AND DISCUSSION

Table 2 contains the results of our various experiments. *P* stands for the *average precision* over the three stances, for each target with a specified algorithm. *R* stands for *recall*, and F_1 is the average F_1-measure.

Table 2: Comparison between the proposed algorithms on SemEval TaskA.

	Algorithm	P	R	F_1
Sent	SENT-BASE	0.41	0.30	0.31
	SENT-CTXT	0.43	0.35	0.37
Stat	SVM-UNIG	0.63	0.62	0.62
	SVM-CTXT	0.58	0.61	0.58
	SVM-EXP	0.69	0.64	**0.66**

SENT-BASE turns out to be a rather unsatisfactory baseline, as it achieves $F_1 = 0.31$. However, SENT-CTXT improves the stance detection to $F_1 = 0.37$. The low scores are due to the weak assumption that sentiment predicts stance. Moreover, the test corpus includes tweets targeting other entities than the *target*: the set whose target is *Hillary Clinton* includes positive mentions to other candidates as well, though it means an *AGAINST* stance towards Hillary Clinton. A sentiment-based approach does not handle this well. Hence, knowing only the topic and the sentiment is insufficient to determine the stance: the target of the sentiment needs be considered as well. In other words, the target of a sentiment in a tweet can differ from the target given by the stance detection task.

SVM-UNIG is a better baseline because it draws upon the training sets (as opposed to SENT-BASE), and reaches $F_1 = 0.62$. One can note that the best *official score* (0.68) on this task was also reached by a SVM (which also included character ngrams in its feature vector) proposed as a baseline by the SemEval organizers (Mohammad et al., 2016).

SVM-CTXT performed rather unsatisfactory. It is mainly due to the small size of the training corpora: the training set has too few elements to cover enough vocabulary of contextosets, thus it is likely that the training set does not cover all of the contextosets possibilities, resulting in SVM-CTXT making prediction on contextosets it has never seen before.

Finally, SVM-EXP shows an improvement, reaching $F_1 = 0.66$ (*Official Score* = 0.65). While it is not better than the results of top competitors, it is comparable to them.

Table 3: Comparison with SemEval competitors, using SemEval official score.

Algorithm:	SVM-EXP	A#1	A#2	A#3
Score:	0.650	0.678	0.673	0.668

In Table 3, we compare our best results to the three best scores obtained by the competitors during the evaluation. Our algorithm SVM-EXP would have been ranked 6^{th} among the 19 competitors: we obtained good results in the *Official Score* benchmark. However, like the other teams, scores are not very high and the accuracy of a prediction is too low to be useful. (Mohammad et al., 2016) proposes an analysis of the results. The top-ranked algorithm (A#1, MITRE) used two recurrent neural networks (RNNs). The first RNN chose the best hashtags on an unlabeled tweet set, and the one second estimated the stance accordingly. The runner-up (A#2, pkudlab) used both a deep convolutionnal neural network and a set of rules, and only used the training data. We are unaware of the technique used by (A#3, TakeLab).

6 CONCLUSION

Stance detection on tweets is a challenging task, because of their shortness, innovative spelling and usage of words. Themes are often implicit and the targets of opinions are not always explicitly mentioned.

In the field of semantic and lexical relatedness, contextonyms and contextosets help to address some of these issues by attempting to disambiguate the words in the tweets. Furthermore, it is possible to

produce contextosets from any kind of dialects or languages, requiring only basic adaptations (e.g. an adapted tokenizer) as well as large amounts of texts.

To show the usefulness of contextosets, we proposed to measure their effect on SemEval stance detection task. We introduced two baselines: a sentiment analyzer, based on SentiWordNet, and a text classifier, based on a SVM whose feature vector is constituted of boolean indicators of unigram presence. In both cases, contextosets increase the global F_1 measure, even though "sentiment" does not seem the best approach on this task.

We believe contextosets have a great potential, and we will continue to explore the possibilities along both sentimental and statistical approaches. Even if our sentiment analyzer failed to predict *positive* tweets of stance *against*, we believe it has the potential to tackle this task. For instance, results may be improved if we enable it to consider the subject of the tweet to grasp not only the sentiment polarity, but also its target. The learning approach may be improved if we use contextosets to disambiguate ambiguous tweets only, and not all of them.

REFERENCES

Anand, P., Walker, M., Abbott, R., Tree, J. E. F., Bowmani, R., and Minor, M. (2011). Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2nd workshop on computational approaches to subjectivity and sentiment analysis*, pages 1–9. Association for Computational Linguistics.

Andreevskaia, A. and Bergler, S. (2008). When specialists and generalists work together: Overcoming domain dependence in sentiment tagging. In *ACL*, pages 290–298.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 2200–2204.

Bradley, M. M. and Lang, P. J. (1999). Affective norms for english words (anew): Instruction manual and affective ratings. Technical report.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.

Feng, Y., Fani, H., Bagheri, E., and Jovanovic, J. (2015). Lexical semantic relatedness for twitter analytics. In *Tools with Artificial Intelligence (ICTAI), 2015 IEEE 27th International Conference on*, pages 202–209. IEEE.

Fernando, S. and Stevenson, M. (2012). Mapping wordnet synsets to wikipedia articles. In *LREC*, pages 590–596.

Gotti, F., Langlais, P., and Farzindar, A. (2013). Translating government agencies tweet feeds: Specificities, problems and (a few) solutions. *NAACL 2013*, page 80.

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA.

Hasan, K. S. and Ng, V. (2013). Extra-linguistic constraints on stance recognition in ideological debates. In *ACL (2)*, pages 816–821.

Hutto, C. J. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth International AAAI Conference on Weblogs and Social Media*.

Hyungsuk, J., Ploux, S., and Wehrli, E. (2003). Lexical knowledge representation with contexonyms. In *9th MT summit Machine Translation*, pages 194–201.

Khan, F. H., Bashir, S., and Qamar, U. (2014). Tom: Twitter opinion mining framework using hybrid classification scheme. *Decision Support Systems*, 57:245–257.

Maynard, D., Bontcheva, K., and Rout, D. (2012). Challenges in developing opinion mining tools for social media. *Proceedings of the@ NLP can u tag# usergeneratedcontent*, pages 15–22.

Maynard, D. and Funk, A. (2011). Automatic detection of political opinions in tweets. In *The semantic web: ESWC 2011 workshops*, pages 88–99. Springer.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

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

Mohammad, S. M., Kiritchenko, S., Sobhani, P., Zhu, X., and Cherry, C. (2016). Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation, SemEval*, volume 16.

Nakov, P., Kozareva, Z., Ritter, A., Rosenthal, S., Stoyanov, V., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 1320–1326.

Palla, G., Dernyi, I., Farkas, I., and Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814–818.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Pennebaker, J. W., Francis, M. E., and Booth, R. J. (2001). Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71:2001.

Perez-Tellez, F., Pinto, D., Cardiff, J., and Rosso, P. (2010). On the difficulty of clustering company tweets. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pages 95–102. ACM.

Ploux, S. and Ji, H. (2003). A model for matching semantic maps between languages (french/english, english/french). *Computational linguistics*, 29(2):155–178.

Rei, M. and Briscoe, T. (2014). Looking for hyponyms in vector space. In *Proceedings of the 18th Conference on Computational Natural Language Learning*, pages 68–77.

Şerban, O. (2013). *Detection and integration of affective feedback into distributed interactive systems*. PhD thesis, Citeseer.

Tan, C.-M., Wang, Y.-F., and Lee, C.-D. (2002). The use of bigrams to enhance text categorization. *Information processing & management*, 38(4):529–546.

Tsytsarau, M. and Palpanas, T. (2012). Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery*, 24(3):478–514.

Wang, R., Zhao, H., Ploux, S., Lu, B.-L., and Utiyama, M. (2016). A bilingual graph-based semantic model for statistical machine translation. In *International Joint Conference on Artificial Intelligence*.

Wiebe, J. and Mihalcea, R. (2006). Word sense and subjectivity. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 1065–1072. Association for Computational Linguistics.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Zesch, T., Müller, C., and Gurevych, I. (2008). Using wiktionary for computing semantic relatedness. In *AAAI*, volume 8, pages 861–866.