Leveraging Paraphrase Labels to Extract Synonyms from Twitter

Maria Antoniak
Pacific Northwest
National Laboratory and
University of Washington
riamarie@uw.edu

Eric Bell
Pacific Northwest
National Laboratory

Fei Xia
University of Washington
riamarie@uw.edu

Abstract

We present an approach for automatically learning synonyms from a corpus of paraphrased tweets. The synonyms are learned by using shallow parse chunks to create candidate synonyms and their context windows, and the synonyms are substituted back into a paraphrase detection system that uses machine translation metrics as features for a classifier. We find a 2.29% improvement in F1 when we train and test on the paraphrase training set, demonstrating the importance of discovering high quality synonyms. We also find 9.8% better coverage of the paraphrase corpus using our synonyms rather than larger, existing synonym resources, demonstrating the power of extracting synonyms that are representative of the topics in the test set.

Introduction

Increasing interest in social media has led to research in efficient analytics for large streams of social data such as blog articles, YouTube comments, and Twitter posts. On Twitter, the volume of tweets makes discovering relevant and new information challenging. Furthermore, the pervasiveness of redundant tweets complicates the task of sifting through the tweets for interesting messages (Zanzotto, Pennacchiotti, and Tsioutsiouliklis 2011).

Finding the new and interesting tweets requires reducing the noise in the data. (Petrović, Osborne, and Lavrenko 2012) showed one solution is to organize tweets into semantically related groups, but this requires the ability to automatically detect paraphrases (determine whether two units, e.g. words, phrases, sentences, have the equivalent semantics). Paraphrase detection can be applied to many other NLP tasks, including summarization, machine translation, and determining textual entailment.

One approach to paraphrase detection includes aligning the words and phrases in the units under comparison. However, when applied to tweets, this approach is hampered by a lack of synonym resources. Twitter contains many words and phrases whose meanings are difficult to determine automatically. Traditional synonym resources such as WordNet (Miller 1995) have limited utility because of the misspellings, acronyms, abbreviations, slang, colloquialisms, and other irregular language usages that are common on Twitter. Some of these phrases share the same meaning but have little or no lexical or semantic (at the word level) overlap. For example, *it would mean the world* and *it would make me so happy* are paraphrases, but the phrases *mean the world* and *make me so happy* are difficult to align; they have no lexical overlap, and the individual words are dissimilar in their standard definitions. These phrases are synonymous only in context.

A further difficulty on Twitter is the great variety of topics represented and the high degree of lexical variation between these topics. Our intuition is that Twitter contains niches of synonyms which might not be applicable to other niches. Named entities and slang that may be only used in context of a certain topic. For example, tweets about a popular celebrity will contain different synonyms from tweets in response to a sporting event (we are unlikely to find synonyms for *dunk* or *make a basket* outside of the topic of basketball). Therefore, synonyms extracted for one topic may not be useful for another. This intuition is supported by the low lexical overlap between even closely related topics on Twitter (see Table 1).

Our goal is to demonstrate the importance of extracting high quality, topic-specific synonyms for Twitter. These synonyms would not only be useful for paraphrase detection. Precise, high-coverage synonyms can be used for reducing the size of the vocabulary, for summarizing content, for query expansion, and for various other tasks.

The remainder of the paper is structured as follows. First, we discuss work related to the task of synonym extraction for Twitter. Next, we describe the datasets used in the current work. Then, we describe our methodology and evaluation techniques, and finally, we present our results, areas for

Topic	Overlap with *Zack Randolph*
Z-Bo	18.88%
Deandre Jordan	17.29%
Scott Brooks	16.71%
Reggie Miller	16.50%
Derek Fisher	15.67%

Table 1: Lexical overlap between tweets on topic *Zach Randolph* and tweets on most similar topics (greatest overlap) from SemEval Corpus.

Source	Type of Synonyms	Number of Synonyms
WordNet	unigrams	117,659 synsets
(Han, Cook, and Baldwin 2012)	unigrams	41,181 pairs
(Xu, Ritter, and Grishman 2013)	n-grams	9,607,561 pairs
PPDB XXXL	n-grams, syntactic rules	169,000,000 rules

Table 2: Comparison of types and sizes of existing synonym lexicons.

future work, and conclusions.

Related Work

Much work has been done on synonym extraction but little explicitly for Twitter. Most previous work has focused on normalization for Twitter, as in (Han, Cook, and Baldwin 2013) and (Chrupala 2014). In this section, we will cover the attempts most relevant to our task of extracting synonyms that can be used for paraphrase detection.

(Ruiz-Casado, Alfonseca, and Castells 2005) describe a traditional approach for synonym extraction. This approach depends on the Distributional Semantics Hypothesis, which states that the meaning of a word is highly correlated with the contexts where it appears, and a corollary of this hypothesis, that synonyms should appear in similar contexts. Therefore, given a pair of words, they label the words as synonyms if the words appear in the same contexts. This approach is unsupervised and requires no data except for a corpus of text. Unfortunately, while this approach achieves 82.50% accuracy on a TOEFL test, it can also result in noise. For example, antonyms are often used in similar contexts, and so the lists of synonyms can be polluted with related but opposite words (Agirre et al. 2009).

Normalization is similar to the task of synonym extraction, but normalization is concerned with reducing noise by normalizing misspellings and lexically similar words. (Han, Cook, and Baldwin 2012) focus on the task of normalization for Twitter. They create a dictionary of one-to-one lexical variants, in which each entry represents a pair of words, one in-vocabulary (IV) and one out-of-vocabulary (OOV). Because of the one-to-one philosophy, no phrases are included in the synonyms and so, for example, no acronyms are normalized. The basic approach is to extract IV-OOV candidate pairs based on their distributional similarity and then re-rank these pairs based on their string similarity.

A second attempt at Twitter normalization was made by (Xu, Ritter, and Grishman 2013), which overcame some of the shortcomings of the previous work. In particular, (Xu, Ritter, and Grishman 2013) did not confine their normalizations to unigrams, and so they were able to normalize acronyms and other phrases. As in a previous approach by (Grigonytė et al. 2010), they first create a corpus of paraphrase pairs and then use these pairs to extract synonyms. To create the paraphrase corpus, they identify tweets that refer to the same event and filter these tweets using Jaccard distance. The tweets were then aligned, and a machine translation system was used to extract normalizations.

Another approach to synonym extraction is the Paraphrase Database (PPDB), a collection of 73 million phrasal paraphrases (Ganitkevitch, Van Durme, and Callison-Burch 2013). Bilingual parallel texts were used to pivot over translations to extract similar phrases. These phrases were then ranked using the cosine distance between vectors of distributional features.

Number of paraphrase pairs	13,063
Average number of tweets per topic	31.8
Percent labeled positively as paraphrases	34%

Table 3: Details of SemEval Corpus training set.

While (Han, Cook, and Baldwin 2012) and (Xu, Ritter, and Grishman 2013) sought to reduce noise and align IV and OOV words, we are interested in comparing noisy data with other noisy data. Of particular interest are synonym phrases, such as colloquialisms, which are lexically divergent. Unlike PPDB, our proposed technique does not rely on bilingual data, and so we can extract synonyms from any corpus without needing to pivot over translations. In this way, we are free to extract synonyms that are specific to the topic of interest.

See Table 2 for a comparison of these existing synonym resources. Although some of these resources have a high number of synonyms, we will show that targeting synonyms to the topic of interest is more important in achieving good coverage.

Data

One challenge in gathering synonyms from Twitter is the lack of labeled data for training a statistical classifier. Because our technique searches for words that would not appear in a traditional dictionary or resource such as WordNet, a training set with equivalent synonyms is difficult to construct.

However, labeled corpora of paraphrases do exist. Specifically, SemEval-15 Task 1[1] includes a training and development set of paraphrase-labeled tweets (Xu et al. 2014) (see Table 3). These tweets were collected for a set of trending topics and labeled using Amazon Mechanical Turk. The labels consist of five votes per pair of tweets, and we follow the recommended voting scheme to determine the binary paraphrase labels. The corpus includes 13,063 pairs of tweets in the training set and 1,902 pairs of tweets in the development set (at the time of writing, the test set was not yet available). We will refer to this corpus as the SemEval Corpus. For clarity, we will refer to the development set as the

[1] http://alt.qcri.org/semeval2015/task1/

test set, since that is how it was used in this work. A portion of the training set was used for development.

Methods

We follow an approach similar to (Ruiz-Casado, Alfonseca, and Castells 2005) in that we search for synonyms that appear in the same context windows within a corpus.

Our methods rely on two intuitions. First, to counteract our lack of training data, we leverage the information given by the paraphrase labels in the SemEval Corpus. The SemEval Corpus is structured such that a series of hypothesis tweets all share the same reference tweet. We make the assumption that the hypothesis tweets are all paraphrases of each other as well as of the reference tweet (the paraphrase labels are transitive). Because synonyms are likely to appear in these groups of paraphrases while antonyms and related but non-synonymous words are not likely to appear, we only search within these groups of paraphrases for our synonyms, rather than across the entire corpus.

Our second intuition is to use chunks from a shallow parser to form our candidate synonyms and context windows. By using these chunks, we hope to align synonyms of varying length and improve performance over simple n-grams, which split the tweet arbitrarily. We use Alan Ritter's Twitter NLP Tools[2] (Ritter et al. 2011) to chunk the tweets.

After using the shallow parser to obtain candidate synonyms, we strip stop words from each chunk. Next, we remove chunks which fail any of the following tests: if the length of the chunk is less than three characters, if the chunk contains more than four unigrams, or if the length of the chunk is greater than seven characters but contains less than two vowels. We also discard chunks that are identical except for a number, e.g. *pls follow* and *pls follow89*. This eliminates some noise, such as nonsense words that contain only consonants and spam containing numbered tweets. We do not apply a frequency threshold because we are particularly interested in rare and unusual synonyms that are specific to the corpus.

Each chunk is surrounded by a left-hand chunk and right-hand chunk (we use 'BOS' and 'EOS' as chunks at the beginning and end of the tweet). These form the context windows which are used to align the candidate synonym with other synonyms. If a chunk occurs at least once in the same context window as another chunk, we label those candidates as synonyms. This low threshold is possible because of our strictness in only searching within groups of paraphrases.

From the resulting synonym chunks, we extract shorter synonyms, using sliding windows of n-grams within the synonym chunks. We follow the same approach as above except that we use n-grams instead of shallow parse chunks. For example, if we have two chunk synonyms *lizzie mcguire* and *lizy mcguire*, we would extract *lizzie* and *lizy* as synonyms.

Evaluation

We test our synonyms by substituting them back into the paraphrase corpus and running a baseline paraphrase detection system. First, we substitute the extracted synonyms back into the SemEval corpus. For each pair of tweets A and B, we search within tweet A for n-grams contained in our synonym list. If an n-gram a is found, we search tweet B for synonyms of a. If we find a synonym b in tweet B and a is not already present in tweet B, we replace b with a.

[2]http://github.com/aritter/twitter_nlp

Example extracted synsets
nets and bulls game, netsbulls game, netsbulls series, nets bulls game
classic, hilarious, fucking crazy, live, good game, priceless, just friggin amazing
sac, 916, sac bitch, sac town
nice, piff, aite
fam david amerson, team redskinsnation, family, redskins, washington redskins
dallas, cowboys
two minutes, 180 seconds, 2 minutes, 180 secs, 2 mins, minute, record 2 minutes, approximately 2 minutes

Table 4: Example synonyms extracted from SemEval Corpus training set.

We use a simplified implementation of (Madnani, Tetreault, and Chodorow 2012) as a baseline paraphrase detection system. This system uses machine translation metrics as features for a supervised classification model. Machine translation metrics were originally developed to test the similarity between computer-generated and human-produced translations. However, we can consider paraphrases as English-to-English translations (or monolingual translations) and use the machine translation metrics to detect the semantic similarity and paraphrase relationship between the sentences. (Madnani, Tetreault, and Chodorow 2012) use a variety of machine translation metrics as features for a statistical classifier. We limit these metrics to Translation Edit Rate (TER) and Translation Edit Rate Plus (TERp) (Snover et al. 2009). TER aligns the words between two candidate sentences, and TERp improves on TER by incorporating synonym resources that can align synonyms and synonym phrases in the cadidate sentences, rather than relying on exact matches.

We use these metrics as features for a statistical classifier. Each pair of tweets is labeled with a binary paraphrase label depending on whether the tweets are paraphrases of each other. The task of predicting new labels could be accomplished with a variety of statistical classifiers, and for our baseline system, we choose to use support vector machines (SVM) (Hearst et al. 1998).

We chose this system because while it performs well on standardized data such as the Microsoft Research Paraphrase Corpus (MSRPC) (Quirk, Brockett, and Dolan 2004), it performs poorly on Twitter. This low performance is due to TERps reliance on WordNet and a paraphrase phrase table which are not representative of the vocabulary on Twitter. Therefore, the system is a good candidate for improvement through Twitter-specific synonyms.

Results

Table 6 displays the results of our system. Variations in train and test sets for the SVM classifier are shown, but the syn-

Example substitutions
i can see tyler wilson getting the starting job in **oakland** i can see tyler wilson getting the starting job in **raiders**
damn why the jets **released** tebow damn why the jets **canned** tebow
jones got a tko at the end of the **1st** jones got a tko at the end of the **first round**
hey stella **pls follow** me xx hey stella **please follow** me xx
that **pacific rim con footage** is a big pile of incredible that **pacific rim trailer** is a big pile of incredible
coutinho is a **super** player coutinho is a **proper** player

Table 5: Example substitutions of synonyms back into the training set.

SVM Train	SVM Test	Synonyms Substituted	Accuracy	F1
train	test	no	74.09	52.33
train	test	yes	7**4.24**	**52.68**
train	train	no	76.01	58.89
train	train	yes	**76.82**	**61.18**
test	test	no	74.07	52.18
test	test	yes	**74.14**	**52.42**

Table 6: Accuracy and F1 scores for paraphrase detection with different combinations of train and test sets with synonyms extracted from the training set.

onyms substituted into these sets are always extracted from the training set. Our results in Table 7 show an improvement of 0.81% in accuracy and 2.29% improvement in F1 when we substitute our synonyms into the training set and run the paraphrase detection system on the same training set. When we run the paraphrase detection system on the test set, we see a smaller improvement: 0.15% in accuracy and 0.35% in F1. We show these results from training and testing on the training set to demonstrate the possible improvement given by high quality synonyms. Because the SemEval test set covers different topics than the training set, it is not surprising that synonyms extracted from the training set boost the training set's score but not the test sets scores.

By leveraging the paraphrase labels, we are able to avoid classic errors such as grouping antonyms or related words, e.g. the days of the week. The synonyms and synonym phrases produced through this method do include some noise, partly due to errors in the shallow parse, but the majority are accurate and specific to the given topics (see Table 4). For example, by examining our synonyms, one can correctly conclude that our corpus includes many tweets about sporting events. Such specific synonyms usually do not appear in other synonym resources, such as WordNet or the PPDB. As another example, when discussing a sports team, *family* and *philadelphia eagles* are synonyms, even though they would not be synonyms outside the topic of the sports team the *Philadelphia Eagles*. Similarly, *wizz* and *wiz kalifa* are synonyms, but unless the corpus used for synonyms extraction includes these variations, they would not be detected. From these results, we see that our method is effective at extracting synonyms for the topics present in the training set.

Synonyms	Accuracy	F1
WordNet baseline	76.01	58.89
Han et al., 2012	76.02	58.93
PPDB XXXL Phrasal	76.10	59.10
Xu et al., 2013	76.15	59.25
Current work	**76.82**	**61.18**

Table 7: Accuracy and F1 scores for paraphrase detection with different synonym lexicons used for substitution when training and testing on the SemEval Corpus training set.

Synonyms Source	Number of Substitutions	Percentage of Substitutions
Han et al., 2012	15	0.1%
Xu et al., 2013	123	1.1%
PPDB XXXL Phrasal	133	1.1%
Current work	**1,262**	**10.9%**

Table 8: Numbers of substitutions on the SemEval training corpus using different synonyms sets.

Table 8 displays the number of synonym substitutions discovered by different synonym lexicons on the SemEval training set. Although the total number of synsets we extract is much smaller than the number of synsets in previous synonym resources (see Table 2), we find a 9.8% increase in coverage when we target our synonyms to the topics of interest. In subsequent tests, we found similar increases in coverage when we used a simple n-gram approach as in (Ruiz-Casado, Alfonseca, and Castells 2005) over a separate corpus of tweets on the same topics, showing that the increase in coverage is a result of a focus on the topics and not merely a reflection of using the same dataset to extract the synonyms. This provides further support for our theory of lexical niches existing on Twitter, and it demonstrates that tailoring synonyms to their topics is crucial.

Future Work

Our next step is to extract synonyms that are more representative of the test sets topics. We have demonstrated that such synonyms are valuable to paraphrase detection, and it may be possible to bootstrap a larger paraphrase corpus, as in (Xu, Ritter, and Grishman 2013), targeted to the test set's topics and use this corpus to extract synonyms. This would remove our dependence on the labeled paraphrase corpus for synonym extraction and make our method more generally applicable.

It may also be possible to build a silver-standard training set. Synonym labels could be extracted from WordNet or other traditional dictionaries, and feature vectors could be constructed from the corpus of interest. This labeled set could be used to train a classifier, which would allow for more subtle feature weighting.

This work also shows potential for disambiguating named entity references. Many of the extracted synonyms are variations on the same named entity. For example, *philly*, *eagles*, and *birdgang* are all synonyms referring to the same sports team, the Philadelphia Eagles. Our system groups these n-

grams and so could be used disambiguate the words and phrases used to refer to the entity.

Conclusion

We have shown that automatically extracted synonyms can be used to improve the results of paraphrase detection on Twitter, if the synonyms are sufficiently representative of the test set. We have further shown that given a corpus of paraphrases, we can extract high-quality synonyms that avoid common errors such as equating antonyms. Finally, our approach outperforms previous synonym resources both in accuracy/F1 for paraphrase detection and coverage on the training set, and it relies on a few simple intuitions and requires no bilingual data or labeled synonyms.

References

Agirre, E.; Alfonseca, E.; Hall, K.; Kravalova, J.; Paşca, M.; and Soroa, A. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 19–27. Association for Computational Linguistics.

Chrupala, G. 2014. Normalizing tweets with edit scripts and recurrent neural embeddings. 680–686.

Ganitkevitch, J.; Van Durme, B.; and Callison-Burch, C. 2013. PPDB: The paraphrase database. In *Proceedings of NAACL-HLT*, 758–764. Atlanta, Georgia: Association for Computational Linguistics.

Grigonytė, G.; Cordeiro, J.; Dias, G.; Moraliyski, R.; and Brazdil, P. 2010. Paraphrase alignment for synonym evidence discovery. In *Proceedings of the 23rd International Conference on Computational Linguistics*, 403–411. Association for Computational Linguistics.

Han, B.; Cook, P.; and Baldwin, T. 2012. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 421–432. Association for Computational Linguistics.

Han, B.; Cook, P.; and Baldwin, T. 2013. Lexical normalization for social media text. *ACM Transactions on Intelligent Systems and Technology (TIST)* 4(1):5.

Hearst, M. A.; Dumais, S. T.; Osman, E.; Platt, J.; and Scholkopf, B. 1998. Support vector machines. *Intelligent Systems and their Applications, IEEE* 13(4):18–28.

Madnani, N.; Tetreault, J.; and Chodorow, M. 2012. Re-examining machine translation metrics for paraphrase identification. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 182–190. Association for Computational Linguistics.

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

Petrović, S.; Osborne, M.; and Lavrenko, V. 2012. Using paraphrases for improving first story detection in news and twitter. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 338–346. Association for Computational Linguistics.

Quirk, C.; Brockett, C.; and Dolan, W. B. 2004. Monolingual machine translation for paraphrase generation. In *EMNLP*, 142–149.

Ritter, A.; Clark, S.; Etzioni, O.; et al. 2011. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1524–1534. Association for Computational Linguistics.

Ruiz-Casado, M.; Alfonseca, E.; and Castells, P. 2005. Using context-window overlapping in synonym discovery and ontology extension. *Proceedings of RANLP-2005, Borovets, Bulgaria* 39:43.

Snover, M. G.; Madnani, N.; Dorr, B.; and Schwartz, R. 2009. Ter-plus: paraphrase, semantic, and alignment enhancements to translation edit rate. *Machine Translation* 23(2-3):117–127.

Xu, W.; Ritter, A.; Callison-Burch, C.; Dolan, W. B.; and Ji, Y. 2014. Extracting lexically divergent paraphrases from Twitter. *Transactions of the Association for Computational Linguistics*.

Xu, W.; Ritter, A.; and Grishman, R. 2013. Gathering and generating paraphrases from twitter with application to normalization. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora (BUCC)*, 121–128. Sofia, Bulgaria: Association for Computational Linguistics.

Zanzotto, F. M.; Pennacchiotti, M.; and Tsioutsiouliklis, K. 2011. Linguistic redundancy in twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 659–669. Association for Computational Linguistics.