Classifying Out-of-vocabulary Terms in a Domain-Specific Social Media Corpus

SoHyun Park,♣ Afsaneh Fazly,◊ Annie Lee,◊ Brandon Seibel,◊ Wenjie Zi◊ and Paul Cook♣
♣ Faculty of Computer Science, University of New Brunswick
◊ VerticalScope, Toronto Canada
sohyun.park@unb.ca, {afazly,alee,bseibel,wzi}@verticalscope.com, paul.cook@unb.ca

Abstract

In this paper we consider the problem of out-of-vocabulary term classification in web forum text from the automotive domain. We develop a set of nine domain- and application-specific categories for out-of-vocabulary terms. We then propose a supervised approach to classify out-of-vocabulary terms according to these categories, drawing on features based on word embeddings, and linguistic knowledge of common properties of out-of-vocabulary terms. We show that the features based on word embeddings are particularly informative for this task. The categories that we predict could serve as a preliminary, automatically-generated source of lexical knowledge about out-of-vocabulary terms. Furthermore, we show that this approach can be adapted to give a semi-automated method for identifying out-of-vocabulary terms of a particular category, automotive named entities, that is of particular interest to us.

Keywords: Lexical acquisition, social media text, out-of-vocabulary words

1. Domain-specific OOV Classification

Out-of-vocabulary terms are more common in social media text than more-conventional text types (Baldwin et al., 2013). Moreover, many domain-specific technical terms are not included in general-purpose dictionaries and lexical resources. Domain-specific social media corpora are therefore particularly rife with out-of-vocabulary terms.

Many natural language processing (NLP) systems for tasks including sentiment analysis and question answering rely on lexical knowledge. In the case that a text being processed contains out-of-vocabulary terms, system performance suffers because lexical knowledge is not available for these words. Much research in NLP has therefore focused on lexical acquisition — automatically learning syntactic or semantic properties of words from corpora (Hearst, 1992; Lin, 1998; Turney and Littman, 2003, for example).

In this work we focus specifically on out-of-vocabulary (OOV) terms in web forum text from the automotive domain. The focus on the automotive domain is motivated by the business interests of VerticalScope, Inc. (the industrial collaborator in this research) in being able to more-intelligently analyze this text. VerticalScope, Inc. is a Canadian company that owns and operates one of the most highly visited automotive networks of online forums. The goal of this research is to automatically classify OOVs as one of a predefined set of domain- and application- specific categories — e.g., automotive named entity (NE), slang, spelling error, foreign term. This automatically-inferred coarse-grained lexical knowledge will later be used to build vocabularies focused on, or excluding, particular types of expressions in an effort to improve topic models (Blei et al., 2003, for example) of automotive web forum text to better understand its contents. This knowledge will also be leveraged in an effort to improve downstream NLP tasks, such as named entity recognition, for this specialized text type.

2. OOVs in Automotive Social Media

For this study, we focused on a random sample of 665 alphanumeric OOVs that 1.) have frequency greater than 1000 in an automotive web forum corpus of roughly 150 million posts from the years 2013 and 2014; 2.) consist of two to ten characters; 3.) do not occur in any of the GNU Aspell English Dictionary version 0.7,[1] a list of automotive terms from Freebase,[2] or a list of automotive acronyms.

These OOVs were manually annotated by a single judge as one of nine OOV categories, described in Table 1. The categories were determined based on the common types of OOVs found in this data. The annotator was a computational linguist with knowledge of the automotive domain, who was not otherwise involved in building the automatic OOV classification system.

A sample of twenty items was also annotated by a second judge, also an author of this paper, who in this case was directly involved in building the automatic OOV classification system. The observed agreement and unweighted Kappa were, 0.65 and 0.55, respectively. We are, however, particularly interested in the NE-AUTO class. These terms include car makes, models, and trims (e.g., *XLT* in *Ford F-150 XLT*), that are not listed in any of the lexical resources considered. Because of the relatively low inter-annotator agreement on the nine-way annotation task, we therefore also considered a two-way annotation task for the categories NE-AUTO and "other" (i.e., the eight other classes). Here the observed agreement and unweighted Kappa were 0.90 and 0.78, respectively, suggesting that human annotators can much more reliably distinguish these two categories than the full set of nine categories.

3. Model

In this preliminary work we consider a supervised approach to OOV classification using the following classes of features.

[1] http://aspell.net/

[2] https://www.freebase.com/

Category	Num. items	Explanation	Examples
AUTO	45	Automotive terms (not NEs)	*defuel*, *rebalance*
DRUG	95	Drug names	*levoxyl*, *nexium*
FOREIGN	47	Non-English terms	*rezeptfrei*, *depuis*
MEASUREMENT	58	Units of measurement	*77k*, *100mph*
NE-AUTO	140	Automotive-related NEs	*ls3*, *volks*
NE-OTHER	41	Non-automotive NEs	*blackhawks*, *diaz*
NOISE	87	Noise, and items that don't fit other categories	*kagvjfcjfx*, *kzvddzfv52*
SLANG	59	Internet slang and non-standard forms	*heyyaa*, *lol2*
SPELLING-ERROR	93	Spelling errors	*youll*, *genericfor*

Table 1: The categories of OOVs, along with an explanation of, and examples of, each.

3.1. Character N-grams

Certain character n-grams are more frequent in some categories than others. For example, spelling errors and non-standard social media forms contain character sequences that are uncommon in standard English due to character deletion or repetition. In drug names, character sequences such as word-final ne and an (e.g., as in *ketamine* and *niaspan*) are particularly common. Character n-grams are often applied in language identification (Lui and Baldwin, 2011, for example) and could therefore be particularly informative for identifying foreign terms. Our first set of features therefore consists of the character n-grams in a given OOV, for n = 1–3.

3.2. Character N-gram Models

Many NE-AUTO OOVs contain character sequences that are rare in standard English (e.g., *ls2* an engine model name). Moreover, many FOREIGN OOVs contain character sequences that are uncommon in English. We construct character-level bigram and trigram models from corpora of English, German, and Spanish. For English we used the Brown Corpus (Francis and Kucera, 1979), while for German and Spanish we used the corresponding versions of the Universal Declaration of Human Rights available through NLTK (Bird et al., 2009). We also built language models for a list of automotive acronyms, and a list of automotive terms from Freebase. For these features, for a given OOV, we calculate its probability under each of these bigram and trigram language models.

3.3. Frequency

We hypothesize that categories such as SLANG and SPELLING-ERROR will tend to be infrequent in well-edited text, and relatively frequent in text types such as social media text. Moreover, categories such as NE-AUTO and AUTO will tend to be relatively frequent in text from that domain — whether social media text or not — and relatively infrequent in other domains. We obtain corpora corresponding to a variety of text types (described in Section 4.1. below). For these features, for a given OOV, we calculate its frequency in each corpus normalized by the total number of tokens in the corresponding corpus.

3.4. Word Embeddings

Word embeddings (e.g., word2vec, Mikolov et al., 2013) are vector representations of words that capture aspects of their syntax and semantics. We hypothesize that the embedding for a given OOV will tend to be close in vector space to OOVs of the same category. For this feature set, we run word2vec on a corpus of web forum text (described in Section 4.1. below), and use the resulting embedding for a given OOV as a feature vector. In the case that an OOV does not have a word embedding (which happens when the OOV does not occur in the corpus, or when it does occur in the corpus but has frequency below a threshold), we represent it as the average of the vectors for all other OOVs that do have word embeddings.

We use the following settings for word2vec: the skipgram model, a vector dimensionality of 200, a window size of 5, and a minimum frequency of 10. To select these parameters, we trained word2vec with a variety of parameter settings for the model (skipgram and cbow), number of dimensions, and window size. We evaluated the vectors obtained on the analogy task of Mikolov et al. (2013); the selected parameters gave the best results on this task.

3.5. Surface Form

We introduce five further features based on observed properties of the surface forms of OOVs.

- We observe that many OOVs in the SLANG and SPELLING-ERROR categories appear to be formed by concatenating two in-vocabulary words (e.g., *eachother*, a spelling error, is formed from *each* and *other*). The first surface form feature is a binary feature that takes the value 1 if a given OOV can be split into a prefix and suffix that each occur in a dictionary (the GNU Aspell English Dictionary), and 0 otherwise.

- Many OOVs in the NOISE and MEASUREMENT categories are relatively shorter than items in the other categories. We hypothesize that word length could be informative of categories of OOVs. The second surface form feature corresponds to the number of characters in a given OOV.

- We observe many OOVs that consist of a combination of letters and digits in the categories NOISE (e.g., *w43y5h6*), SLANG (e.g., *high5*), and NE-AUTO (e.g., *m8000*). The third surface form feature feature represents the wordshape of a given OOV. Contiguous sequences of consonants, vowels, and digits are mapped

Corpus	Num. docs	Num. tokens	Num. OOVs from dataset
Wikipedia	4.9M	1.7B	340
Twitter	98.9M	1.2B	474
Forums	80.0M	5.0B	657

Table 2: The number of documents, tokens, and OOVs from our dataset, in each corpus.

to the symbols *c*, *v*, and *d*, respectively. For example, *high5* is represented as *cvcd*.

- We note many OOVs in NOISE, SLANG, and SPELLING-ERROR that have many repeated letters, such as *mmm* and *loong*. The fourth surface form feature represents whether a given OOV contains two consecutive repeated characters.
- Many spelling errors are within a small edit distance (often one or two) of their correction. The minimum of the edit distance between an OOV and any in-vocabulary word could therefore be particularly informative as to whether that OOV is a SPELLING-ERROR. For the final surface form feature we calculate the minimum edit distance between a given OOV and any word in a dictionary (again, GNU Aspell).

4. Materials and Methods

4.1. Corpora

Three corpora were used for this research:

Forum Posts from a variety of English web forums such as `talkford.com`, `watchfreeks.com`, and `samsunggalaxyreviews.com` corresponding to verticals such as automotive, collectibles, and technology.

Wikipedia A dump of English Wikipedia from 1 September 2015.

Twitter A sample of English tweets collected from the Twitter Streaming APIs[3] from November 2014 to March 2015.

The Wikipedia corpus was pre-processed to remove meta-data, such as wiki markup, using WikiExtractor.[4] The forum corpus was similarly pre-processed to remove forum tags. The Wikipedia corpus was tokenized using the Stanford CoreNLP tokenizer (Manning et al., 2014). The forum and Twitter corpora were tokenized using a tokenizer developed for tweets, adapted from (O'Connor et al., 2010). Table 2 shows the number of documents, tokens, and OOVs from our dataset, in each corpus.

These corpora were used to calculate the frequency features (Section 3.3.). The forums corpus was used for training word2vec to calculate the word embeddings features (Section 3.4.).

4.2. Experimental Setup

10x10-fold stratified cross-validation experiments were carried out on the dataset of OOVs using the features described above with a maximum entropy classifier.[5] We considered both a nine-way classification task (i.e., classifying OOVs according to the categories in Table 1) and a two-way classification task for the classes NE-AUTO and "other" (i.e., the remaining eight categories). We considered each feature set on its own, as well as combinations of feature sets. For each experiment we calculated macro-averaged precision, recall, and F1 score, as well as accuracy. As a point of comparison, we further considered a most-frequent class baseline.

5. Experimental Results

Macro-averaged precision, recall, and F1 score, as well as accuracy, are shown in Table 3 for the most-frequent class baseline, each feature set individually, all feature sets combined, and ablative experiments in which we consider all-but-1 feature set, for each feature set, for the nine-way classification task.

The character n-gram, word embedding, and surface form features all substantially outperform the baseline, in terms of all evaluation metrics, while the character n-gram models and frequency features perform on par with the baseline. This suggests that character n-grams in OOVs, distributional information captured by word embeddings, and the linguistic knowledge captured by the surface form features are all highly informative for OOV classification in automotive web forum text. In future work we intend to explore the use of additional corpora, in particular domain-specific texts from non-social media text types, to further explore the impact of frequency in OOV classification.

The combination of all features (shown as [A+B+C+D+E] in Table 3) performs roughly on par with the best individual feature set, the word embeddings. We further explore combinations of features in ablative experiments, in which we consider all feature sets but one, holding out each feature set in turn.

The classifier using all feature sets except the frequency features ([A+B+D+E] in Table 3) performs best in terms of all evaluation metrics. That these features (all feature sets excluding the frequency features) improve over any individual feature set indicates that they carry complementary information about OOV categories. Moreover, this further suggests that the frequency features do not carry information about OOV categories as they are currently formulated. The relatively low performance when the word embedding features are omitted ([A+B+C+E] in Table 3) reinforces the power of these features for this task.

We now consider precision, recall, and F1 score for each class, using the best-performing features for the nine way classification task (all feature sets except the frequency features, i.e., [A+B+D+E]). Results are shown in Table 4. The F1 scores for DRUG and MEASUREMENT, 0.892 and 0.843,

[3] `https://dev.twitter.com/`

[4] `https://github.com/attardi/wikiextractor`

[5] We also considered random forests, a linear support vector machine (SVM), and an SVM using a radial basis function kernel, however, we saw little difference in results, and so only describe and report results for maximum entropy here.

Method	Precision	Recall	F1 score	Accuracy
Most-frequent class baseline	0.023	0.111	0.039	0.211
[A] Characater n-grams (1-3)	0.390	0.373	0.380	0.413
[B] Character n-gram models	0.023	0.111	0.039	0.211
[C] Frequency	0.023	0.111	0.039	0.211
[D] Word embeddings	0.649	0.599	0.622	0.643
[E] Surface form	0.390	0.400	0.394	0.446
[A+B+C+D+E]	0.643	0.603	0.622	0.649
[B+C+D+E]	0.649	0.602	0.624	0.646
[A+C+D+E]	0.640	0.605	0.622	0.648
[A+B+D+E]	**0.650**	**0.609**	**0.628**	**0.654**
[A+B+C+E]	0.429	0.422	0.424	0.469
[A+B+C+D]	0.614	0.582	0.597	0.629

Table 3: Macro-averaged precision, recall, and F1 score, as well as accuracy, for the baseline, each feature set, and combinations of features for the nine-way classification task.

Method	Precision	Recall	F1 score
DOMAIN-TERM	0.586	0.459	0.489
DRUG	0.899	0.945	0.892
FOREIGN	0.675	0.744	0.673
MEASUREMENT	0.903	0.883	0.843
NE-OTHER	0.466	0.200	0.367
NE-AUTO	0.633	0.802	0.703
NOISE	0.727	0.610	0.645
SLANG	0.468	0.433	0.462
SPELLING-ERROR	0.524	0.499	0.502

Table 4: Precision, recall, and F1 score, for each class in the nine-way classification task, using the best performing feature combination from Table 3.

respectively, are relatively high. We are, however, particularly interested in NE-AUTO, because of our need to identify automotive named entities (such as car makes and models) that are not in our current resources. Here the F1 score is somewhat lower, 0.703, with a precision and recall of 0.633 and 0.802, respectively.

We now consider whether we can improve performance on NE-AUTO by re-formulating the classification task as a 2-way task. Here we consider a two-way classification task for NE-AUTO VS OTHER, i.e., all other classes. Results are shown in Table 5. In this case the most-frequent class is OTHER. As such, the most-frequent class baseline achieves a precision, recall, and F1 score of 0 because no items are classified as NE-AUTO. The word embeddings are again the best of the individual feature sets in terms of recall, F1 score and accuracy. However, in this case the surface form features achieve the highest precision. The best F1 score for the two-way task is achieved using all feature sets except the character n-gram features ([B+C+D+E] in Table 5). In this case the F1 score (0.676) is somewhat lower than the F1 score for the NE-AUTO class using the best overall features for the nine-way task (0.703, Table 4). However, there is a precision–recall trade-off. The precision for NE-AUTO for the best overall features for the two-way task (0.725) is higher than that for the nine-way task (0.633, Table 4), although the recall is lower (0.644 vs. 0.802).

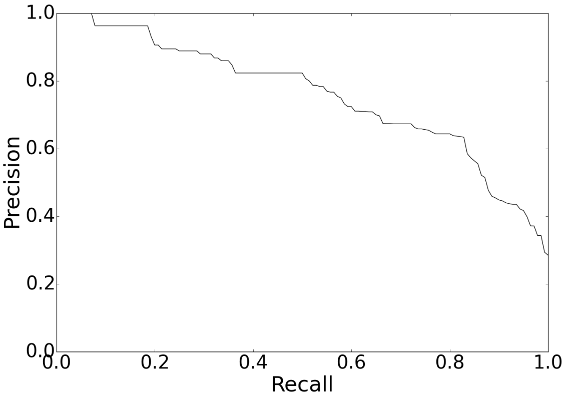

Figure 1: Interpolated precision–recall curve for NE-AUTO.

We further consider precision and recall for the two-way classifier by examining a precision–recall curve. For the two-way classifier using all features except the character n-grams, we rank all items in the dataset by the probability of the NE-AUTO class. The precision–recall curve is shown in Figure 1. Precision remains relatively high, roughly 0.8, for recall values up to 0.5. This suggests that this ranking could be useful for semi-automatic identification of NE-AUTO terms, where lexicographers could further analyze highly-ranked items.

6. Conclusions

In this paper we considered the problem of domain-specific OOV classification in web forum text, focusing on the automotive domain. We demonstrated that supervised methods trained on features based on word embeddings for OOVs are highly informative for this task, and can be complemented by information from features based on linguistic knowledge of common properties of OOVs. The coarse-grained OOV categories that we predict could serve as a preliminary, automatically-generated source of lexical knowledge about OOVs. Moreover, we showed that such methods could be used to rank OOVs to produce a semi-

Method	Precision	Recall	F1 score	Accuracy
Most-frequent class baseline	0.000	0.000	0.000	0.789
[A] Characater n-grams (1-3)	0.499	0.222	0.292	0.788
[B] Character n-gram models	0.000	0.000	0.000	0.786
[C] Frequency	0.000	0.000	0.000	0.789
[D] Word embeddings	0.714	0.619	0.656	0.862
[E] Surface form	**0.775**	0.154	0.288	0.814
[A+B+C+D+E]	0.734	0.583	0.609	0.866
[B+C+D+E]	0.725	**0.644**	**0.676**	**0.873**
[A+C+D+E]	0.741	0.586	0.618	0.867
[A+B+D+E]	0.735	0.567	0.615	0.869
[A+B+C+E]	0.510	0.257	0.332	0.793
[A+B+C+D]	0.728	0.557	0.586	0.863

Table 5: Precision, recall, and F1 score, for the NE-AUTO class, as well as accuracy, for the baseline, each feature set, and combinations of features, for the two-way classification task.

automated approach to identifying automotive named entities among the OOVs.

In future work we intend to apply this approach to classifying OOVs to build vocabularies focused on, or excluding, particular kinds of OOVs to be used by other NLP applications, such as topic modeling. We further intend to apply this knowledge in downstream NLP tasks, such as named entity recognition, for domain-specific web forum text.

7. Acknowledgments

This research is funded by the Natural Sciences and Engineering Research Council of Canada.

8. Bibliographical References

Baldwin, T., Cook, P., Lui, M., MacKinlay, A., and Wang, L. (2013). How noisy social media text, how diffrnt social media sources? In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP 2013)*, pages 356–364, Nagoya, Japan.

Bird, S., Loper, E., and Klein, E. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc., Sebastopol, CA.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Francis, W. N. and Kucera, H., (1979). *Manual of Information to accompany A standard Corpus of Present-Day Edited American English, for use with Digital Computers*. Brown University.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the Fourteenth International Conference on Computational Linguistics (COLING 1992)*, pages 539–545, Nantes, France.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (ACL/COLING 1998)*, pages 768–774, Montreal, Canada.

Lui, M. and Baldwin, T. (2011). Cross-domain feature selection for language identification. In *Proceedings of the Fifth International Joint Conference on Natural Language Processing (IJCNLP 2011)*, pages 553–561, Chiang Mai, Thailand.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, USA.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, USA.

O'Connor, B., Krieger, M., and Ahn, D. (2010). TweetMotif: Exploratory search and topic summarization for Twitter. In *Proceedings of the 4th International Conference on Weblogs and Social Media (ICWSM 2010)*, pages 384–385, Washington, USA.

Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346.