Improving In-Domain Data Selection For Small In-Domain Sets

Mohammed Mediani, Joshua Winebarger, Alexander Waibel

Karlsruhe Institute of Technology
Karlsruhe, Germany
firstname.lastname@kit.edu

Abstract

Finding sufficient in-domain text data for language modeling is a recurrent challenge. Some methods have already been proposed for selecting parts of out-of-domain text data most closely resembling the in-domain data using a small amount of the latter. Including this new "near-domain" data in training can potentially lead to better language model performance, while reducing training resources relative to incorporating all data.

One popular, state-of-the-art selection process based on cross-entropy scores makes use of in-domain and out-of-domain language models. In order to compensate for the limited availability of the in-domain data required for this method, we introduce enhancements to two of its steps.

Firstly, we improve the procedure for drawing the out-of-domain sample data used for selection. Secondly, we use word-associations in order to extend the underlying vocabulary of the sample language models used for scoring. These enhancements are applied to selecting text for language modeling of talks given in a technical subject area.

Besides comparing perplexity, we judge the resulting language models by their performance in automatic speech recognition and machine translation tasks. We evaluate our method in different contexts. We show that it yields consistent improvements, up to 2% absolute reduction in word error rate and 0.3 Bleu points. We achieve these improvements even given a much smaller in-domain set.

1. Introduction

The need for in-domain data in machine learning is a well-established problem and should be well motivated in previous papers (e.g [1]). We briefly observe, however, that across domains system performance is tied to the similarity between training and testing data. The testing data used for guiding system development is almost synonymous with in-domain data. It follows directly that training data should also resemble the in-domain as closely as possible. In-domain data however is also almost always the most limited kind. This necessitates supplementing it with out-of-domain or non-domain-specific data in order to achieve satisfactory model estimates.

In this paper we consider the training of language models for speech recognition and machine translation of university lectures, which are very domain-specific. Typically this means adapting existing systems to a new topic. Perhaps unique to this application is that the in-domain data for lectures is normally of a very small size. A one-hour lecture may produce under a thousand utterances and roughly ten thousand words. The necessity of rapid system development and testing in this context encourages us to limit training data size. What we want, then is a way to reduce large amounts of data and at the same time improve its relevance. Ideally we would also be able to do so using only a very small amount of in-domain data.

We improve the work of [2] by drawing a better representative sample of out-of-domain data and language model (LM) vocabulary. However, more centrally, we extend the work of [2] by using a word-association based on a broad definition of similarity to extend these language models. With this extension, we do not compare solely the exact matching words from in-domain and out-of-domain corpora, but also their semantically associated words. These semantic associations can be inferred, as in the example of this paper through the use of pre-existing non-domain-specific parallel and/or monolingual corpora, or through hand-made thesauri. Then with a small amount of in-domain data we use the aforementioned extended language models to rank and select out-of-domain sentences.

1.1. Previous Work

The starting point and reference of our work is that found in [2], which is to our knowledge one of the most recent and popular methods in a series of methods on data selection [3, 4, 5]. Their approach assumes the availability of enough in-domain data to train a reasonable in-domain LM, which is used to compute a cross-entropy score for the out-of-domain sentences. The sentence is also scored by another, out-of-domain LM resulting from a similar-sized random out-of-domain sample. If the difference between these two scores exceeds a certain threshold the sentence is retained, the threshold being tuned on a small heldout in-domain set. This approach can be qualified as one based on the perplexity of the out-of-domain data. The in-domain data used in [2] is the EPPS corpus, which contains more than one million sentences. This stands in contrast to the lecture case with very specific domains and very limited data sizes. The

authors report their results in terms of perplexity, for which their technique outperforms a baseline selection method by twenty absolute points. Their approach has been shown to be effective for selecting LM training data, at least from the perspective of a Statistical Machine Translation (SMT) system with a specific domain task [6, 7, 8]. We note that the main task of these systems was to translate TED talks.[1] The work in [2] was extended to parallel data selection by [9, 10]. However, the last work concludes that the approach is less effective in the parallel case.

The approach of differential LM scores used in the aforementioned papers has a long history in the information retrieval (IR) domain [11, 12]. However, only unigram language models are considered in the context of IR, since the order in this task is meaningless.

Enriching the LM capability by incorporating word relationships has also been proposed in IR and is referred to as a *translation model* therein [13, 14].[2] More closely related to our approach, [15] uses word similarities to extend LMs in all orders. They show that extended LMs with properly computed word similarities significantly improve their performance at least in a speech recognition task.

1.2. Area of Application

The translation of talks and lectures between natural languages has gained attention in recent years, with events such as the International Workshop on Spoken Language Translation (IWSLT) sponsoring evaluations of lecture translation systems for such material as TED talks. From the perspective of Automatic Speech Recognition (ASR), talks and lectures are an interesting domain where the current state of the art can be advanced, as the style of speaking is thought to lie somewhere between spontaneous and read speech.

As noted previously, university lectures in particular are very domain-specific and thus in-domain data tends to be quite limited. The typical approach for language modeling in such a scenario is to include as much data as possible, both in- and out-of-domain, and allow weighted interpolation to select the best mixture based on some heldout set. However, if a satisfactory method could be found to choose only those parts of the out-of-domain set most similar to the in-domain set, this would reduce the amount of necessary LM training data. Not only would this save training time, it would also produce LMs that are smaller and possibly more adapted to the task at hand.

We perform text selection using variations of our technique and train language models on the resulting selected data. These LMs are then evaluated in terms of their perplexity on a heldout set, the word-error-rate of a speech recogniser, and an SMT system using the LM. We also apply the technique of [2] to our selection task as a reference.

[1] `http://www.ted.com`

[2] Note that we will use the terms "translation model" and "lexicon" interchangeably throughout the paper.

1.3. Paper structure

The remainder of the paper is structured as follows. In section 2 we describe the theory behind our enhancements to the standard selection algorithm. First, we discuss our method of intelligently selecting the out-of-domain LM used for cross-entropy selection. Next, we discuss our experiments with a more careful selection of the cross-entropy in-domain and out-of-domain language model vocabularies. In section 3.1 we introduce our association-based approach. We describe how we compute lexicons and how we use them to extend the cross-entropy language models. The results of our experiments are presented in section 5. We end the paper with section 6 in which we draw conclusions and discuss future work.

2. Enhancements

2.1. Drawing an out-of-domain representative sample

In the cross-entropy method of [2] previously described in Section 1.1, the out-of-domain LM is taken simply as a random sample of the larger out-of-domain data upon which we do selection, *OD*. However, randomly-drawn text may represent both in-domain as well as out-of-domain data (*OD*). The out-of-domain LM should instead represent the kind of data which we seek to exclude from our selection. Since the in-domain data should be the furthest from the latter kind of data, we reasoned that the in-domain LM could be used to intelligently select the data for the out-of-domain LM. We do this by first scoring the sentences in *OD* with the in-domain LM for perplexity (with a closed vocabulary). As some of our data in *OD* comes from web crawls, the sentences with the highest perplexity are mainly "junk" coming from automatic text processors and/or converters. The sentences with the lowest perplexity are mostly in the in-domain set. Therefore we specify some range around the median perplexity (m) as being a legitimate region from which to select sentences for the out-of-domain LM. In our case we chose $m \pm 0.5m$ with m being the median perplexity. Then for our out-of-domain LM we randomly draw an appropriate number of sentences from this range. The probability of any particular sentence being drawn is proportional to its corresponding perplexity.[3]

2.2. Vocabulary selection

Intuitively, we could think of vocabulary words as indicators of the importance of a sentence. Words occurring with high frequency in both in- and out-of-domain data sets would be of lower interest. In contrast, words frequently encountered in the in-domain only indicate that the sentence is of high importance. It was not clear to us whether the words which are common in the out-of-domain only would be a negative indicator. That is why we experimented with different ways

[3] For the weighted random sampling without replacement, we use the algorithm described in [16]

for choosing the vocabulary on which the LMs are based. The first vocabulary is taken as the intersection of the in- and out-of-domain vocabularies $V_1 = voc\{ID\} \cap voc\{OD\}$. The second vocabulary incorporates the first and adds those words which occur with high frequency in the in-domain source only. This is $V_2 = V_1 \cup hf\{ID\}$. The third incorporates the second (and consequently the first,) adding those high-frequency words occurring only in the out-of-domain LM dataset. Thus $V_3 = V_1 \cup hf\{OD\}$ A visual representation of this scheme is depicted in figure 1.

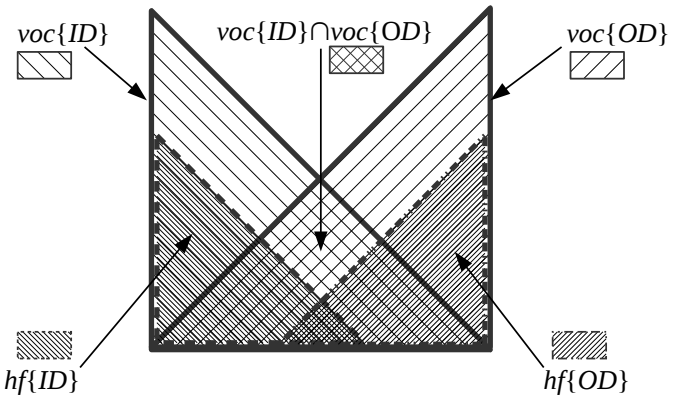

Figure 1: Diagrammatic representation of vocabularies of in- and out-of-domain sources

3. Extended Cross-Entropy Selection

In this section, we present our approach to create the word associations resulting in a lexicon quantifying the strength of relationships between vocabulary words and non-vocabulary words. First, the theoretical motivation for this kind of association is presented. Then the technical details on how our lexicon was built are discussed. Finally, the unigram LM extension is explained.

3.1. From bilingual word alignments to monolingual word associations

It is noteworthy that the lexical word-associations could be derived in many ways. These include manually hand-crafted thesauri (e.g. WordNet [17]) or automatically learned from monolingual corpora [18]. In this work, most of our experiments are based on lexicons derived form freely available parallel corpora, since we already dispose of relevant parallel data and computational tools to perform such a task.

Our lexicon derivation is based on the following assumption: In a perfectly aligned parallel corpus, words from the source language aligned to the same target word should be lexically related. Consequently, in creating a lexicon for a language (say, German) we infer associations between the (German) source words from their aligned target words (say, in English.) The association between two source words is proportional to the alignment probabilities relating them to the common target word.

Based on this assumption, we would like to estimate relationship strength (the so-called translation table) for pairs of words. One word of such a pair, the vocabulary word, is found in the LM vocabulary (and hence in the in-domain sample). The selection of this vocabulary is explained in Section 2.2. The other word comes from the source side (i.e. German) of the parallel corpus but is not present in the LM vocabulary.

Given a vocabulary word v and a non-vocabulary word w, the association $\mathrm{t}(w \mid v)$ is estimated as follows:

$$\begin{aligned} \mathrm{t}(w \mid v) &= \frac{\Pr(w, v)}{\Pr(v)} \\ &= \sum_z \frac{\Pr(z)\Pr(w, v \mid z)}{\Pr(v)} \\ &\approx \sum_z \frac{\Pr(z)\Pr(w \mid z)\Pr(v \mid z)}{\Pr(v)} \\ &= \sum_z \Pr(w \mid z)\Pr(z \mid v) \end{aligned} \tag{1}$$

In the second line of Equation (1), we rewrote the probability expression by introducing the aligned words z from the target side (i.e English) as a latent variable. In the third line, we simplified the expression in the previous line by assuming that source words are independent when conditioned on the target words.

3.2. Lexicon creation

We create our lexicon from automatically aligned parallel corpora (EPPS, NC, and Common Crawl). The corpora are preprocessed by removing obvious tokens which would not contribute to associating words such as numbers and punctuation marks. Then we use the Giza++ toolkit to train the IBM3 alignments in both directions (i.e German $\rightarrow$ English and English $\rightarrow$ German). We then symmetrize the resulting alignments using the intersection heuristic [19]. That is to say, we retain only alignment points which appear in both directions. An additional symmetrizing step we perform is removing links corresponding to a negative association.[4]

The resulting alignments allow us to compute the terms $\Pr(w \mid z)$ and $\Pr(z \mid v)$ in Equation (1) and therefore the lexicon.[5] The probabilities from this lexicon will be used to induce a likelihood for the words which do not occur in the original vocabulary of our LMs used for computing cross-entropy scores. We discuss this LM extension in Section 3.4.

[4]Two words x and y are negatively associated if $\Pr(x, y) < \Pr(x)\Pr(y)$ [20].

[5]In machine translation literature, the terms $\Pr(w \mid z)$ and $\Pr(z \mid v)$ are referred to as *Lexical Translation Models* (not to be confused with the model referred to as Translation Model in IR).

3.3. Associations from monoligual corpora

A more attractive approach to computing associations between words would be by exploiting monolingual resources. These are available in much more important quantities for any language compared to their parallel counterparts. We explored this approach by using the cosine similarity between word vectors returned by `word2vec` [21] to infer word associations. For each vocabulary word we include the 10 most similar non-vocabulary words in the resulting lexicon. The similarity score between a vocabulary word v and a non-vocabulary word w is computed as follows:

$$\mathrm{Sim}\,(w, v) = \frac{\mathbf{w} \cdot \mathbf{v}}{\|\mathbf{w}\|\|\mathbf{v}\|} + 1$$

where $\mathbf{w}$ and $\mathbf{v}$ are the word vectors associated with w and v respectively.

Then, the association $\mathrm{t}(w \mid v)$ is obtained by normalizing the similarity scores, as follows:

$$\mathrm{t}\,(w \mid v) = \frac{\mathrm{Sim}\,(w, v)}{\sum_{w'} \mathrm{Sim}\,(w', v)}$$

3.4. Extension of LMs

According to the cross-entropy selection, the out-of-vocabulary (OOV) words will have only a small effect on a sentence score. This is due to the fact that they are mapped to <unk> (the unknown word,) and therefore the probability returned from one model (e.g. the in-domain) cancels its counterpart from the other (e.g. the out-of-domain.)[6] Consequently, including more "important" words in the model with a realistic likelihood would conceivably make our model more robust.

To extend the LM with knowledge from the lexicon, we add to the unigram order those words which in the lexicon are associated with the LM vocabulary words. Therefore, these new unigrams can contribute to evaluating the sentence probabilities by the back-off mechanism. We found that the rate of backing-off to these new words is about 20%. The integration of the new unigrams is performed as follows. First, we discount the probabilities of the vocabulary words to free some a priori fixed mass (say $1 - m_0$.) Afterwards, each word added from the lexicon receives a share from m_0 proportional to two factors. The first factor is the LM probability of the associated vocabulary words. The second factor is the strength of the lexicon association connecting the out-of-vocabulary word to the in-vocabulary words. Note that m_0 is a tunable parameter. In our experiments, we found setting $m_0 = \Pr(<\text{unk}>)$ to be optimal.

Formally speaking, the probability of observing the word w given that the word sequence $w*$ is expressed as follows:

$$\Pr\,(w \mid w^*) = \begin{cases} m_0 \Pr_{LM}\,(w \mid w^*) & \text{if } |w^* w| > 0 \\ (1 - m_0) \sum_{v:|w^* v|>0} \mathrm{t}\,(w \mid v) \Pr_{LM}\,(v \mid w^*) & \text{otherwise} \end{cases}$$

[6] This effect will mostly be a penalization. In practice, the probability of <unk> is larger in the out-of-domain model

where w^* is an arbitrary sequence of words, possibly empty (for unigrams); $\Pr_{LM}$ is the original back-off LM probability; $|x^*|$ is the number of times the sequence x^* appears in the text; and t is the association table associating a vocabulary word v to a non-vocabulary word w. This procedure results in a new LM whose vocabulary is a superset of the original vocabulary. However, in most of this work we applied this extension at the unigram level only and hence kept the number of higher order n-grams unchanged.

4. Experimental Design

4.1. Data sources

For our out-of-domain data, we used a collection of monolingual German-language text corpora from various sources. This corpus totals around 37 million sentences and 0.67 billion tokens. We call this set of corpora *OD*. A table summarising these sources is given in table 1.

Type	Sentence count	Token count
News	11M	204M
Blog	3M	45M
Webcrawls	18M	345M
Parliamentary transcripts	256K	3.4M
Speeches and talks	6.8K	164K
Other sources	1.2K	18K
Total	37M	670M

Table 1: *Summary of monolingual out-of-domain text data used as a basis for data selection, which we term* OD

For bi-word association and lexicon training, we used a German-English parallel corpus we term *PC*. This consists of the public parallel corpora distributed for the WMT evaluation campaign [22] totaling 3.3 million lines of parallel text.

An in-domain corpus was available totaling 11 thousand lines and 237 thousand tokens, taken as mixture of transcriptions of several university lectures. We call this corpus *DEV*. Another similar-sized set from the same domain was held out in order to evaluate the perplexity of the resulting LMs.

For the purpose of computing ASR word error rates (WER), we took as a basis 16 hours of transcribed in-domain talk and lecture recordings from our in-house resources. The transcriptions for this set, composed of 13 thousand lines with 168 thousand tokens, were used as a set of held-out in-domain text for testing the perplexity of our language models. This held-out set is named *TEST*.

From the 16 hours of audio we randomly selected one hour on which to test the ASR. We call it *WERTEST*.

4.2. Selection Process

Our process of creating a set of selected texts from *OD* proceeded in several steps. Given *DEV* and *OD* we created an in-domain LM and out-of-domain LM. In our experiments with association-based scoring we extend the in-domain and out-of-domain LMs with information from our

lexicon. Next, scores were computed for each line in each source in *OD*. We then ranked all candidate lines across sources according to their score and retained only the top $K\%$ of candidates to carry over into the selected corpus *SEL*.

Our baseline experiments focused on creating selections from the base set, varying the top $K\%$ retained between 1% and 100%. After creating the set *SEL*, we performed some text normalisation such as compound word splitting.

German, the test language of our experiments, is known for use of compound words. As this makes contributes to a high out-of-vocabulary rate in ASR, a compound-splitting algorithm is typically employed in this field. For example "Entscheidungsfunktion" is split into "Entscheidungs+ Funktion." This algorithm requires a list of sub-words and selects the best split by maximizing the sum of the squared sub-word-lengths [23]. The *TEST* and *DEV* corpora are preprocessed using this technique, whereas as the alignment texts for the lexicon training are not. This necessitated the application of compound splitting after selection and prior to LM training.

5. Results

In this section, we compare the results of the different techniques mentioned in the previous sections (enhancements and extensions).

In our first sets of experiments as shown in Tables 2 and 3, we perform selection using a reasonably-sized in-domain set, *DEV*, with around eleven thousand sentences. [7] In Table 2 we report perplexity values of the LMs on *TEST*. For each selection technique we show the results of retaining either the top 1, 2, 5, 6, or 10% of sentences. The first row in the table is our baseline consisting of the state-of-the-art cross-entropy method of [2]. The improvements gained from the enhancements are shown in the second row. The remaining rows are related to applying the extension in different ways.

As shown in the third row, we apply the extension to the in-domain LM in the process of drawing the out-of-domain sample as explained in Section 2.1. For this we used only the high-frequency in-domain vocabulary, $hf\{ID\}$ as shown in Figure 1. After that, we retrain both the in- and out-of-domain LMs without extension. This configuration is referred to as "Extended Enhancements" (seen in the table as "Ext. Enhancements.")

In the fourth row we show the results of our "Extension" configuration. This configuration applies the extension only to our final in- and out-of-domain selection LMs (i.e., no extension was applied while drawing the out-of-domain sample), using the approach described in Section 3.4.

Finally, the previous two extensions are effectively combined. This means that we apply two independent extensions: we extend the in-domain LM in order to draw the out-of-domain representative and then we extend both in- and out-of-domain LMs for selection. We see this configuration, "Double Extension," on the fifth row of the table.

Table 3 shows WER resulting from using a subset of these LMs in a recognition task.

Technique	% Retained Sent. (ppl)				
	1	2	5	6	10
Moore, et al	222.7	202.4	190.3	190.0	190.5
Enhancements	211.9	195.4	185.3	184.5	185.9
Ext. Enhancements	208.1	192.9	183.4	183.3	185.0
Extension	206.2	191.9	183.0	182.5	184.4
Double Extension	203.0	189.1	181.3	181.0	183.3
Reference	% Retained Sent. (ppl)				
	100				
No selection	301.9				

Table 2: *Perplexity on TEST of the LMs selected using a reasonable in-domain set*

Technique	% Retained Sent. (WER)		
	1	5	10
Moore, et al	30.5	29.1	29.5
Enhancements	30.2	28.7	28.9
Double Extension	29.9	28.2	28.5
Reference	% Retained Sent. (WER)		
	100		
No selection	29.9		

Table 3: *Word error rate on WERTEST of LMs selected using a reasonable in-domain set*

In our second set of experiments, we simulated the case of hard conditions on the availability of in-domain data. We used a very small set of only one thousand sentences for our in-domain set as follows. First we split *DEV* into two parts, each part begin scored using the other. Then we merged them and selected the top-scoring one thousand sentences. This way, we assume that the resulting small set would be concentrated on the dominating topic of the whole set. The results of using this small in-domain set are summarized in Table 4.

We see that in the case of the small in-domain set, our method outperforms the baseline of [2] by between 40 and 60 perplexity points, and up to 2 percentage points absolute in terms of WER. For the reasonably-sized in-domain set, using enhancement alone gives larger gains than the incremental gains made by applying extension as well. For the small in-domain set, applying extension adds incremental gains comparable to the initial gains from enhancement.

Furthermore, we tested some of our selected data in a machine translation task. This is a phrase-based statistical system, where the translation model is trained on EPPS, NC, TED and BTEC English-German parallel corpora. It was tuned and tested on portions of a computer science lecture. The development set is around one thousand pairs whereas the test set is about two thousand. The weights of the log-linear model were tuned for a system using an LM trained on

[7] It is noteworthy that even this reasonably-sized in-domain set is less than 1% of the size of the in-domain set used in [2].

Technique	% Retained Sent. (ppl)		% Retained Sent. (WER)	
	5	10	5	10
Moore, et al	297.3	256.3	32.4	31.3
Ext. Enhancements	267.0	237.7	31.7	30.8
Double Extension	230.1	216.4	30.2	29.8

Table 4: *Perplexity and WER on TEST and WERTEST of LMs selected using a reduced in-domain set*

a completely different set. These were then kept unchanged for all tested models. The results of the translation experiments are shown in Table 5. Both enhancement and extension always outperformed the baseline. However for the cases of 10 and 20 percent retained sentences, the extension did not bring any additional gain.

Technique	% Retained Sent. (BLEU)		
	5	10	20
Moore, et al	13.24	13.04	12.84
Enhancements	13.47	13.19	13.06
Extension	13.52	13.16	13.00
Reference	% Retained Sent. (BLEU)		
	100		
No selection	12.47		

Table 5: BLEU scores for translation results

Finally, we performed some additional experiments in order to examine the extension in all ngram orders and the usage of associations induced from monolingual corpora. Table 6 shows the corresponding results. The first row repeats the last one in Table 2. The second row shows the results of a full extension, where we use the same principle as detailed in Section 3.4 in order extend words of the LM. However, here we extend all orders from 1 through 4 unlike the previous experiments where we only extended the unigrams. The results of monolingual-based associations are shown in the third row. In this case, the association is equivalent to the cosine similarity between word vectors (as explained in Section 3.3.) These vectors are computed using a large corpus (29 million sentences and 0.4 billion tokens). To do so, we use `word2vec` with *continuous bag of words* as the learning algorithm [21].[8] The size of the vectors is set to 500 and the context window to 10. Words appearing less than 5 times are discarded and the number of iterations used is 15.

It follows from these last experiments that both full extension and `word2vec` associations have no important effect on the performance. However, these can be considered as baselines for future experiments as they lack thorough hyper parameter tuning.

[8] `http://code.google.com/p/word2vec/`

Technique	% Retained Sent. (ppl)		
	1	5	10
Double Extension (only unigrams)	203.0	181.3	183.3
Full extension	203.0	181.4	183.4
`word2vec` associations	203.3	181.7	183.6
Reference	% Retained Sent. (ppl)		
	100		
No selection	301.9		

Table 6: *Perplexity on TEST of additional experiments*

6. Conclusion

We presented several extensions and enhancements to the state-of-the-art in-domain data selection method of [2]. Our techniques bring consistent improvements to the performance of the LM, given enough similarity between the test set and the set used for selection. Improvement is noticeable for a reasonably-sized in-domain set and it is quite more noticeable still for very small in-domain sets, where in terms of perplexity we substantially outperform the state-of-the-art. In both ASR and SMT scenarios, our techniques proved efficient by aggressively reducing the size of the training data. At the same time, they consistently improved the system's performance or in the worst case kept it unchanged.

While the automatically computed associations are cheaper to obtain, their hand-made counterparts are likely to be more accurate. Consequently, we plan to perform a comparison between these two for English, as it disposes of the largest hand-made thesaurus (WordNet).

It might be questioned why the associations used throughout this paper were inferred from general domain corpora, as this may lead to undesirable associations for a specific domain. Therefore, we would like to explore the effect of a pre-selection process over the data used to compute the association lexicon.

For the very small in-domain data sets, we think that better results could be obtained if one follows a bootstrapping strategy. That is, we repeatedly perform selection and add the best scoring sentences to the in-domain set and use the resulting set as the in-domain set for the next run.

We found both full extension and `word2vec` associations to be more expensive than the alignment-based unigram extension. Full extension suffers from a combinatorial explosion when the vocabulary size is reasonable. `word2vec` associations, on the other hand, are very slow to compute since we need to test each pair of words. We think we could improve this by performing the extension on a carefully selected subset from the vocabulary.

Another question we need to look into is the way we convert cosine similarities of `word2vec` into appropriate associations. The values we get from our current implementation are almost uniform. This might explain why this approach could not outperform the alignment-based associations, in spite of a much larger training corpus.

Lastly we close by noting that the tools developed for

lexicon creation are freely available on Github.[9]

7. Acknowledgments

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 287658. The authors would also like to thank Jan Nieheues, Yuqi Zhang, and Ahmed Abdelali for their review and constructive comments.

8. References

[1] I. H. Daumé and D. Marcu, "Domain Adaptation for Statistical Classifiers," *J. Artif. Int. Res.*, vol. 26, no. 1, pp. 101–126, May 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1622559.1622562

[2] R. C. Moore and W. D. Lewis, "Intelligent Selection of Language Model Training Data," in *ACL (Short Papers)*, 2010, pp. 220–224.

[3] D. Klakow, "Selecting articles from the language model training corpus," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, vol. 3. IEEE, 2000, pp. 1695–1698.

[4] S.-C. Lin, C.-L. Tsai, L.-F. Chien, K.-J. Chen, and L.-S. Lee, "Chinese language model adaptation based on document classification and multiple domain-specific language models," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[5] J. Gao, J. Goodman, M. Li, and K.-F. Lee, "Toward a unified approach to statistical language modeling for Chinese," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 1, no. 1, pp. 3–33, 2002.

[6] N. Durrani, B. Haddow, K. Heafield, and P. Koehn, "Edinburghs machine translation systems for European language pairs," in *Proceedings of the Eighth Workshop on Statistical Machine Translation*, 2013, pp. 112–119.

[7] J. Wuebker, M. Huck, S. Mansour, M. Freitag, M. Feng, S. Peitz, C. Schmidt, and H. Ney, "The RWTH Aachen machine translation system for IWSLT 2011," in *Proceedings of IWSLT*, 2011, pp. 106–113.

[8] T.-L. Ha, T. Herrmann, J. Niehues, M. Mediani, E. Cho, Y. Zhang, I. Slawik, and A. Waibel, "The KIT translation systems for IWSLT 2013," in *Proceedings of IWSLT*, 2013.

[9] A. Axelrod, X. He, and J. Gao, "Domain adaptation via pseudo in-domain data selection," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 355–362.

[10] S. Mansour, J. Wuebker, and H. Ney, "Combining translation and language model scoring for domain-specific data filtering," in *Proceedings of IWSLT*, 2011, pp. 222–229.

[11] J. Lafferty and C. Zhai, "Document Language Models, Query Models, and Risk Minimization for Information Retrieval," in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '01. New York, NY, USA: ACM, 2001, pp. 111–119. [Online]. Available: http://doi.acm.org/10.1145/383952.383970

[12] W. Kraaij and M. Spitters, "Language Models for Topic Tracking," in *Language Models for Information Retrieval*, B. Croft and J. Lafferty, Eds. Kluwer Academic Publishers, 2003. [Online]. Available: http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-153-22-33670504-detailsPage%253Dppmmedia%257Ctoc%257Ctoc,00.html

[13] A. Berger and J. Lafferty, "Information Retrieval As Statistical Translation," in *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '99. New York, NY, USA: ACM, 1999, pp. 222–229. [Online]. Available: http://doi.acm.org/10.1145/312624.312681

[14] G. Cao, J.-Y. Nie, and J. Bai, "Integrating Word Relationships into Language Models," in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '05. New York, NY, USA: ACM, 2005, pp. 298–305. [Online]. Available: http://doi.acm.org/10.1145/1076034.1076086

[15] I. Dagan, L. Lee, and F. C. N. Pereira, "Similarity-based models of word cooccurrence probabilities," *Machine Learning*, vol. 34, no. 1-3, pp. 43–69, 1999. [Online]. Available: http://dx.doi.org/10.1023/A:1007537716579

[16] P. S. Efraimidis and P. G. Spirakis, "Weighted random sampling with a reservoir," *Information Processing Letters*, vol. 97, no. 5, pp. 181– 185, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S002001900500298X

[17] G. A. Miller, "Wordnet: A lexical database for english," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995. [Online]. Available: http://doi.acm.org/10.1145/219717.219748

[9] https://github.com/medmediani/pdict

[18] P. D. Turney, P. Pantel, *et al.*, "From frequency to meaning: Vector space models of semantics," *Journal of artificial intelligence research*, vol. 37, no. 1, pp. 141–188, 2010.

[19] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, ser. NAACL '03. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003, pp. 48–54. [Online]. Available: http://dx.doi.org/10.3115/1073445.1073462

[20] S. Evert, "The statistics of word cooccurrences," Ph.D. dissertation, University of Stuttgart, 2004.

[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[22] "ACL 2014 Ninth Workshop on Statistical Machine Translation, Results and Collected Judgments," http://www.statmt.org/wmt14/translation-task.html, accessed: 2014-07-20.

[23] T. Marek, "Analysis of german compounds using weighted finite state transducers," *Bachelor thesis, University of Tübingen*, 2006.