RECURRENT NEURAL NETWORK LANGUAGE MODELING FOR CODE SWITCHING CONVERSATIONAL SPEECH

Heike Adel[1]*, Ngoc Thang Vu*[1]
Franziska Kraus[1]*, Tim Schlippe*[1]*, Haizhou Li*[2]*, Tanja Schultz*[1]

[1]Cognitive Systems Lab, Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT)
[2]School of Computer Engineering, Nanyang Technological University (NTU), Singapore
heike.adel@student.kit.edu, thang.vu@kit.edu

ABSTRACT

Code-switching is a very common phenomenon in multilingual communities. In this paper, we investigate language modeling for conversational Mandarin-English code-switching (CS) speech recognition. First, we investigate the prediction of code switches based on textual features with focus on Part-of-Speech (POS) tags and trigger words. Second, we propose a structure of recurrent neural networks to predict code-switches. We extend the networks by adding POS information to the input layer and by factorizing the output layer into languages. The resulting models are applied to our task of code-switching language modeling. The final performance shows 10.8% relative improvement in perplexity on the SEAME development set which transforms into a 2% relative improvement in terms of Mixed Error Rate and a relative improvement of 16.9% in perplexity on the evaluation set which leads to a 2.7% relative improvement of MER.

Index Terms: code-switching, recurrent neural network language model

1. INTRODUCTION

Code-switching speech is defined as speech that contains more than one language ('code'). The switch between languages may happen between or within an utterance. It is a common phenomenon in many multilingual communities where people of different cultures and language background communicate with each other [1]. For the automated processing of spoken communication in these scenarios, a speech recognition system must be able to handle code switches. Usually, speech recognition systems are monolingual and that is why the task appears to be very difficult to solve. Another challenge is the lack of bilingual training data. While there have been promising research results in the area of acoustic modeling, only few approaches so far address code-switching in the language model. Recently, it has been shown that recurrent neural network language models (RNNLMs) improve perplexity and error rates in speech recognition systems in comparison to traditional n-gram approaches [2, 3, 4]. One reason for that is their ability to handle longer contexts. Furthermore, the integration of additional features as input is rather straight-forward due to their structure. In this paper, we propose a recurrent neural network language model applied for code-switching. We extend its traditional structure by integrating features into the input layer and by factorizing the output layer using language information. Our experimental results demonstrate that this approach leads to significant improvements in terms of perplexity which transform into decent error rate reductions. Figure 1 illustrates our code-switching system.

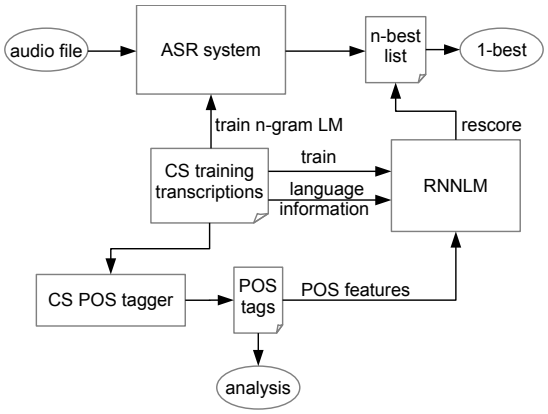

Fig. 1. Overview: our code-switching system

The remainder of the paper is organized as follows: Section 2 reports on previous research in the area of code-switching and language modeling using recurrent neural networks. Section 3 describes the SEAME corpus. Section 4 analyzes the corpus and focuses on the detection of trigger events for code-switching. Section 5 explains our approach of recurrent neural network language modeling for the code-switching task. In section 6, we present our experiments and results. The study is concluded in Section 7.

2. RELATED WORK

Linguistic analyses of the code-switching phenomenon helps to better understand the task and challenges and thus, might help to create an appropriate language model. Hence, various studies on code-switching are described. Furthermore, recent developments in the research on recurrent neural network language models are presented.

2.1. The Code-Switching Phenomenon

In [5, 6, 7], it is observed that code-switches only occur at positions in an utterance where they do not violate the syntactical rules of the involved languages. While [8] argues that the code-switching points are indeterminant because the code-switching decision is entirely up to the individual speakers, [9] discovers some code-switching patterns across speakers. It is shown that speakers mainly switch to another language for nouns and object noun phrases. Therefore, the most frequent switch points are between determiners and nouns and

between verb phrases and object noun phrases.
In [10], different machine learning algorithms (for instance the Naive Bayes Classifier) trained on textual features are used to predict code-switching points. The features include word form, language identity, Part-of-Speech tags and the position of the word relative to the phrase. [11] compares four different kinds of n-gram language models to handle code-switching. It is discovered that a class-based model which clusters all foreign words into their POS classes achieves the best performance. In our previous work [12], we created additional training data for the code-switching task using machine translation. The interpolation of the artificial data with the original code-switching transcriptions outperforms the baseline model.

2.2. Language Modeling Using Recurrent Neural Networks

In the last years, neural networks have been used for a variety of tasks. [2] introduced a refined form of neural networks for the task of language modeling. The so-called recurrent neural networks are able to handle long-term contexts since the input vector does not only contain the current word but also the previous output from the neurons in the hidden layer. It is shown that these networks outperform traditional language models such as n-grams which only contain very limited histories. In [3], the network is extended by factorizing the output layer into classes to accelerate the training and testing processes. Recently, further information has been added to the neural network. [4] augments the input layer to model features such as topic information or Part-of-Speech tags.

2.3. Our Contribution

In this work, we extend the recurrent neural network language modeling toolkit [13] for code-switching. Our analyses which are presented in section 4, show that textual features such as words and Part-of-Speech tags might predict code-switching points. To model this in the structure of our network, we add POS tags to the input layer and the set of all possible languages to the output layer. This leads to the following computation of probabilities as described in section 5: Based on the current word, the current features, and the history of words and features, the probability for the succeeding language is computed. Then, the probability for the next word is computed given the language. This approach which utilizes the factorization of the output layer models the results of the analyses of the code-switching phenomenon. Our experiments demonstrate a significant improvement in terms of perplexity. Moreover, it is shown that rescoring n-best lists with our code-switching language model outperforms the baseline system.

3. SEAME CORPUS

SEAME (South East Asia Mandarin-English) is a conversational Mandarin-English code-switching speech corpus recorded from Singaporean and Malaysian speakers [14]. The recordings consist of spontanously spoken interviews and conversations. Since the publication in [14], the corpus has been extended to about 63 hours of audio data. For this task, all hesitations are deleted and the transcribed words are divided into four categories: English words, Mandarin words, particles (Singaporean and Malaysian discourse particles) and others (other languages). These categories are used as language information in our neural networks. The average number of code-switches between Mandarin and English is 2.6 per utterance. The duration of monolingual segments is very short: More than 82% English and 73% Mandarin segments last less than 1 second, while the average duration of English and Mandarin segments is only 0.67 seconds and 0.81 seconds respectively. In total, the corpus contains 9,210 unique English and 7,471 unique Mandarin vocabulary words. It is divided into three disjoint sets (training, development and test set). The data is assigned based on several criteria (gender, speaking style, ratio of Singaporean and Malaysian speakers, ratio of the four categories, and the duration in each set). Table 1 lists the statistics of the SEAME corpus in these sets.

Table 1. *Statistics of the SEAME corpus*

	Train set	Dev set	Eval set
# Speakers	139	8	10
Duration(hours)	59.2	2.1	1.5
# Utterances	48,040	1,943	1,018
# Token	525,168	23,776	11,294

4. PREDICTION OF CODE-SWITCHES

Similar to the investigations summarized in section 2.1, we perform an analysis of textual features that trigger code-switches on the SEAME data corpus. We concentrate on words and Part-of-Speech tags because an analysis in [15] showed that those are the most important trigger events. We rank them according to their code-switching rate. The code-switching rate is calculated by the frequency of occurences preceeding code-switching points divided by the frequency in the entire text. We considered only those words and POS tags which appear more than 1000 times in the text, corresponding to more than 2% of the entire word tokens.

4.1. Trigger Words

We analyze which words occur frequently immediately in front of code-switching points. Table 2 shows the top five Mandarin and the top five English words preceeding a code-switching point.

Table 2. Mandarin and English trigger words for code-switching points

word	frequency	CS-rate
那个(that)	5261	53.43 %
我的(my)	1236	52.35 %
那些(those)	1329	49.44 %
一个(a)	2524	49.05 %
他的(his)	1024	47.75 %
then	6183	56.25 %
think	1103	37.62 %
but	2211	36.23 %
so	2218	35.80 %
okay	1044	34.87 %

4.2. Part-of-Speech (POS) tags as trigger

Since code-switching speech consists of more than one language, the task of assigning POS tags to words cannot be solved using a traditional monolingual tagger. Hence, we created a POS tagger for code-switching text data [15]. We determine Mandarin as the matrix language (main language of an utterance) and English as the embedded language (the other language) for the SEAME corpus. Three or more consecutive words of the embedded language are called language islands. Those language islands are passed to the Part-of-Speech tagger of the embedded language. The remaining part is

tagged by the tagger of the matrix language. The idea is to provide chunks which are monolingual or contain only a short part of the embedded language to obtain as much context as possible. We use the Stanford log-linear POS tagger for Chinese and English as described in [16, 17]. The tags are derived from the Penn Treebank POS tag set for Chinese and English [18, 19].
An analysis shows that most English words are falsely tagged as nouns by the Chinese tagger. To avoid subsequent errors in the determination of trigger POS tags, we add a postprocessing step to the tagging process: We select all English words that are no language islands and pass them to the English Part-of-Speech Tagger. Then we replace the POS tags of the Chinese tagger with these new tags. After having tagged the code-switching text, we are able to select those tags that possibly predict code-switching points. Two analyses are made as shown in table 3. First, we consider only those tags that appear in front a code-switching point from Mandarin to English. Second, we investigate the tags predicting a code-switching point from English to Mandarin. Table 3 shows that code-switching points are most often triggered by determiners in Mandarin and by nouns in English. This seems reasonable since it is possible that a Mandarin speaker switches for the noun to English and immediately afterwards back to Mandarin. It corresponds to previous investigations as described in section 2.

Table 3. Mandarin and English POS that trigger a code-switching point

Tag	meaning	frequency	CS-rate
DT	determiner	11276	40.44 %
DEG	associative 的	4395	36.91 %
VC	是	6183	25.85 %
DEC	的 in a relative-clause	5763	23.86 %
M	measure word	2612	23.35 %
NN	noun	49060	49.07 %
NNS	noun (plural)	4613	40.82 %
RB	adverb	21096	31.84 %
JJ	adjective	10856	26.48 %
CC	coordinating conjunction	4400	24.05 %

5. RECURRENT NEURAL NETWORK LANGUAGE MODELING FOR CODE-SWITCHING

In this section, we describe the original version of the RNNLM toolkit [13] and our extension to it. Figure 2 illustrates our extension. Nevertheless, the vector names in the following description can be found in it as well. Vector $w(t)$ forms the input of the recurrent neural network. It represents the current word using 1-of-N coding. Thus, its dimension equals the size of the vocabulary. Vector $s(t)$ contains the state of the network. It is called 'hidden layer'. The network is trained using back-propagation through time (BPTT), an extension of the back-propagation algorithm for recurrent neural networks. With BPTT, the error is propagated through recurrent connections back in time for a specific number of time steps t. Hence, the network is able to remember information for several time steps. The matrices U, V and W contain the weights for the connections between the layers. These weights are learned during the training phase. Moreover, the output layer is factorized into classes to accelerate the training and testing processes. Every word belongs to exactly one class. The classes are formed during the training phase depending on the frequencies of the words. Vector $c(t)$ contains the probabilities for each class and vector $w(t)$ provides the probabilities for each word given its class. Hence, the probability $P(w_i|history)$ is computed as shown in equation 1.

$$P(w_i|history) = P(c_i|s(t))P(w_i|c_i, s(t)) \quad (1)$$

The following part describes the recurrent neural network language model which we developed for code-switching.

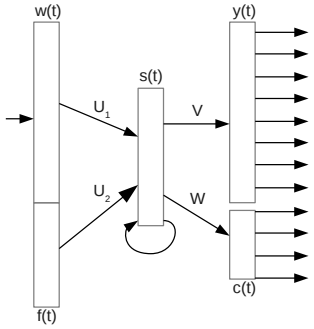

Fig. 2. RNNLM for code-switching
(based upon a figure in [3])

In our extension, the classes of the output layer do not depend on the frequencies of the words but on their languages. We use the language categorization described in section 3. Therefore, our model consists of four classes: One class for all English words, one for all Mandarin words, one for other languages and one for particles. This conforms to the code-switching task because first, the probability of the next language is computed and then the probability of each word given the language. Furthermore, we extend the input layer by concatenating vector $w(t)$ with vector $f(t)$ which provides features corresponding to the current word. According to the analysis described in section 4, we use POS tags as features. We do not use words as feature input for the network because trigger words are implicitly modeled by vector $w(t)$. Vector $f(t)$ consists of 67 elements since the Mandarin words in the vocabulary of the SEAME transcriptions are assigned to 31 POS tags and the English words to 34 POS tags. In addition, the words classified as other languages and the particals form own classes. For each word, a relationship to its POS tag is established. Thus, during the training and testing phases, not only the current word is activated but also its feature. Because the POS tags are integrated into the input layer, they are also propagated into the hidden layer and back-propagated into its history $s(t)$. Thus, not only the previous feature is stored in the history but also all features several time steps in the past. In equation 1, the term $P(c_i|s(t))$ computes the next language c_i using not only information about previous words, but also about previous features.

6. EXPERIMENTS AND RESULTS

In this section, we present the experimental results achieved with our speech recognition system developed for the code-switching task.

6.1. Code-Switch ASR System

Based on the SEAME corpus, we developed a speech recognition system (ASR) as described in [12]. This two-pass system applies first a speaker independent acoustic model which is trained with bottleneck features [20]. The second one is developed by applying Speaker Adaptive Training (SAT) with Feature Space Adaptation

(FSA). To adapt to the code-switching problem and improve accuracy, language identity information is integrated into the decoding process using a multistream approach [21]. To obtain a dictionary, the CMU English [22] and Mandarin pronunciation dictionaries [23] are merged into one bilingual pronunciation dictionary. The number of English and Mandarin entries in the lexicon is 135k and 130k respectively. Additionally, we apply several rules from [24] which might delete or change a phone to generate pronunciation variants for Singaporean English. On the language model side, the SRI Language Modeling Toolkit [25] is used to build trigram language models (LMs) from the SEAME training transcriptions containing all words of the transcriptions. These models are interpolated with two monolingual language models that were created from 350k English sentences from NIST and 400k Mandarin sentences from the GALE project which had been collected from online newspapers. The vocabulary of 30k entries contains all words in the transcriptions and the most frequent words in the monolingual corpora.
Furthermore, characteristics of code-switching from the SEAME training transcriptions are analyzed and additional code-switching text is generated artificially as described in [12]. The resulting language model has a perplexity of 483.9 and an out-of-vocabulary (OOV) rate of 1.21% on the SEAME development set transcriptions. This baseline system achieves an error rate of 35.5% MER on the SEAME development set.

6.2. Language Modeling with the Standard RNNLM

A recurrent neural network without classes serves as a baseline system. It is trained using the code-switching transcriptions. The size of the hidden layer is set to 50 and the BPTT algorithm ran in a block mode with a block size of ten for at least five steps. These parameters were tuned on the development set. This baseline language model has a RNN-perplexity of 246.60 on the development set and of 287.88 on the evaluation set.

6.3. Language Modeling with Code-Switch RNNLMs

A recurrent neural network with factorization of the output layer is developed (*RNNLM + OF*). All other parameters stay the same as in the baseline system to ensure comparability. The classes used are languages: Each English word is mapped to one class and each Mandarin word to another. The third class contains the Mandarin particles and the forth class all other words. This approach reaches a perplexity of 239.64 on the development set and 269.71 on the evaluation set. Hence, the computation of the words depending on their languages improves the performance of the language model in terms of perplexity.
Another network is trained by extending the input layer with POS tags (*RNNLM + FI*). This achieves a perplexity of 233.50 on the development set and 268.05 on the evaluation set. Apparently, the *RNNLM + FI* system outperforms the *RNNLM + OF* system.
Finally, a network is generated with a combination of both techniques (*RNNLM + FI + OF*). The resulting perplexity is 219.85 on the development set and 239.21 on the evaluation set which gives a relative improvement of 10.8% on the development set and of 16.9% on the evaluation set. It outperforms both the *RNNLM + OF* and the *RNNLM + FI*.

6.4. ASR Experiments Using N-best Rescoring

We finally present the performance of each model in terms of mixed error rate when using it for rescoring. In these experiments, we rescore the 100-best lists of our ASR system with different settings for language model weights (lz) and word insertion penalties (lp). Equation 2 shows how the score for each hypothesis is computed. $|w|$ denotes the number of words in the hypothesis and λ the interpolation weight of the recurrent neural network language model. In our experiments, λ is set to 0.5.

$$\begin{aligned} score_{lm} &= \lambda \cdot score_{rnnlm} + (1-\lambda) \cdot score_{n-gram} \\ score &= lz \cdot score_{lm} + score_{am} + lp \cdot |w| \end{aligned} \tag{2}$$

As performance measure, we have established the Mixed Error Rate (MER) which applies word error rates to English and character error rates to Mandarin [12]. It is the weighted average over all English and Mandarin parts of the speech recognition output. By applying character based error rates to Mandarin, the performance does not depend on the word segmentation algorithm for Mandarin. Thus, the performance can be compared across different segmentations. In this case, we used a manual word segmentation.
The code-switching language model (*RNNLM + FI + OF*) achieves the best result with a mixed error rate of 34.7% on the development set and an error rate of 29.2% on the evaluation set. This is an improvement of 2% and 2.7% relative to the baseline system as summarized in table 6.4. (However, this improvement is not statistically significant on the evaluation set compared to the RNNLM baseline.)

Table 4. PPL- and MER-results of different models

Model	PPL dev set	PPL eval set	MER dev set	MER eval set
3-gram	-	-	35.5 %	30.0 %
RNNLM Baseline	246.60	287.88	35.6 %	29.3 %
RNNLM + OF	239.64	269.71	34.9 %	29.4 %
RNNLM + FI	233.50	268.05	34.8 %	29.3 %
RNNLM + FI + OF	219.85	239.21	34.7 %	29.2 %

OF: output factorization, FI: feature integration

We perform an analysis on the SEAME development set to investigate why the *RNNLM + FI + OF* performs better than the standard trigram model. The analysis shows that the trigram model recognizes 1889 code-switching points (41.11%) correctly, whereas the *RNNLM + FI + OF* detects 1990 code-switches (43.31%) correctly. In addition, the *RNNLM + FI + OF* outperforms the trigram model on monolingual segments. On English segments, it achieves a word error rate (WER) of 49.07%, while the trigram model has a WER of 50.21%. On Mandarin segments, the character error rates are 30.32% and 30.90% respectively.

7. CONCLUSIONS

This paper presents our latest investigation on language modeling for conversational Mandarin-English code-switching speech. We showed that particular words and Part-of-Speech tags trigger code-switches more frequently than others. We presented an extension of the standard recurrent neural network for the code-switching task. We used language information to factorize the output layer and integrated Part-of-Speech tags into the input layer. Our experimental results show that this novel RNNLM outperforms the standard RNNLM in both the perplexity and the mixed error rate. In terms of perplexity, the *RNNLM + FI + OF* achieves a relative improvement of 10.8% relative on the development set and 16.9% on the evaluation set. Regarding the MER, the final performance shows 2% relative improvement on the SEAME development set and 2.7% relative on the evaluation set.

8. REFERENCES

[1] P. Auer, *Code-switching in conversation*, Routledge, 1999.

[2] T. Mikolov, M. Karafiát, L. Burget, J. Cernocky, and S. Khudanpur, "Recurrent neural network based language model," in *Proceedings of Interspeech*, 2010.

[3] T. Mikolov, S. Kombrink, L. Burget, JH Cernocky, and S. Khudanpur, "Extensions of recurrent neural network language model," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5528–5531.

[4] Y. Shi, P. Wiggers, and C.M. Jonker, "Towards recurrent neural networks language models with linguistic and contextual features," *Interspeech*, 2012.

[5] S. Poplack, *Syntactic structure and social function of code-switching*, vol. 2, Centro de Estudios Puertorriqueños,[City University of New York], 1978.

[6] E.G. Bokamba, "Are there syntactic constraints on code-mixing?," *World Englishes*, vol. 8, no. 3, pp. 277–292, 1989.

[7] P. Muysken, *Bilingual speech: A typology of code-mixing*, vol. 11, Cambridge University Press, 2000.

[8] P. Auer, "From codeswitching via language mixing to fused lects toward a dynamic typology of bilingual speech," *International Journal of Bilingualism*, vol. 3, no. 4, pp. 309–332, 1999.

[9] S. Poplack, "Sometimes i'll start a sentence in spanish y termino en español: toward a typology of code-switching1," *Linguistics*, vol. 18, no. 7-8, pp. 581–618, 1980.

[10] T. Solorio and Y. Liu, "Learning to predict code-switching points," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2008, pp. 973–981.

[11] J.Y.C. Chan, PC Ching, T. Lee, and H. Cao, "Automatic speech recognition of cantonese-english code-mixing utterances," in *Proceedings of Interspeech*, 2006.

[12] N.T. Vu, D.C. Lyu, J. Weiner, D. Telaar, T. Schlippe, F. Blaicher, E.S. Chng, T. Schultz, and H. Li, "A first speech recognition system for mandarin-english code-switch conversational speech," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4889–4892.

[13] T. Mikolov, S. Kombrink, A. Deoras, L. Burget, and J. Cernockỳ, "RNNLM–recurrent neural network language modeling toolkit," in *Proceedings of the 2011 ASRU Workshop*, 2011, pp. 196–201.

[14] D.C. Lyu, T.P. Tan, E.S. Chng, and H. Li, "An analysis of a mandarin-english code-switching speech corpus: Seame," *Age*, vol. 21, pp. 25–8, 2010.

[15] I.T. Schultz, P. Fung, and C. Burgmer, "Detecting code-switch events based on textual features," 2010.

[16] K. Toutanova, D. Klein, C.D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Association for Computational Linguistics, 2003, pp. 173–180.

[17] K. Toutanova and C.D. Manning, "Enriching the knowledge sources used in a maximum entropy part-of-speech tagger," in *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*. Association for Computational Linguistics, 2000, pp. 63–70.

[18] N. Xue, F. Xia, F.D. Chiou, and M. Palmer, "The penn chinese treebank: Phrase structure annotation of a large corpus," *Natural Language Engineering*, vol. 11, no. 2, pp. 207, 2005.

[19] M.P. Marcus, M.A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of english: The penn treebank," *Computational linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[20] N.T. Vu, W. Breiter, F. Metze, and T. Schultz, "An investigation on initialization schemes for multilayer perceptron training using multilingual data and their effect on asr performance," *Interspeech*, 2012.

[21] J. Weiner, N.T. Vu, D. Telaar, F. Metze, T. Schultz, D.-C. Lyu, E.-S. Chng, and H. Li, "Integration of language identification into a recognition system for spoken conversations containing code-switches," in *SLTU*, 2012.

[22] "CMU pronouncation dictionary for english," `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`.

[23] R. Hsiao, M. Fuhs, Y.C. Tam, Q. Jin, and T. Schultz, "The CMU-InterACT 2008 mandarin transcription system," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2008.

[24] W. Chen, Y. Tan, E. Chng, and H. Li, "The development of a singapore english call resource," *Oriental COCOSDA, Nepal*, 2010.

[25] A. Stolcke et al., "SRILM-an extensible language modeling toolkit," in *Proceedings of the international conference on spoken language processing*, 2002, vol. 2, pp. 901–904.