

Game State Retrieval with Keyword Queries

Atsushi Ushiku
Kyoto University
ushiku@ar.media.kyoto-u.ac.jp

Hiroataka Kameko
The University of Tokyo
kameko@logos.t.u-tokyo.ac.jp

Shinsuke Mori
Kyoto University
forest@i.kyoto-u.ac.jp

Yoshimasa Tsuruoka
The University of Tokyo
tsuruoka@logos.t.u-tokyo.ac.jp

ABSTRACT

There are many databases of game records available online. In order to retrieve a game state from such a database, users usually need to specify the target state in a domain-specific language, which may be difficult to learn for novice users. In this work, we propose a search system that allows users to retrieve game states from a game record database by using keywords. In our approach, we first train a neural network model for symbol grounding using a small number of pairs of a game state and a commentary on it. We then apply it to all the states in the database to associate each of them with characteristic terms and their scores. The enhanced database thus enables users to search for a state using keywords. To evaluate the performance of the proposed method, we conducted experiments of game state retrieval using game records of *Shogi* (Japanese chess) with commentaries. The results demonstrate that our approach gives significantly better results than full-text search and an LSTM language model.

KEYWORDS

Search of Nonlinguistic Data, Shogi, Symbol Grounding

ACM Reference format:

Atsushi Ushiku, Shinsuke Mori, Hiroataka Kameko, and Yoshimasa Tsuruoka. 2017. Game State Retrieval with Keyword Queries. In *Proceedings of ACM SIGIR Conference, Tokyo, Japan, August 2017 (SIGIR'17)*, 4 pages.
DOI: 10.475/123.4

1 INTRODUCTION

Recently, the increasing amounts of nonlinguistic data are raising the demands for nonlinguistic information retrieval. As the interface of such retrieval systems, natural language is potentially very useful because it would save users from

learning (often complex) formats of search queries. Currently, however, retrieval systems that accept natural language queries for nonlinguistic data are available only for a few domains such as images [4] and videos [13].

One example of nonlinguistic data is stock charts. A stock trader may want to search for a stock chart similar to his current situation in order to predict what will happen next. Another example is records of games. A chess player may want to practice by using past game records for reference. Building a natural language-based search system for such nonlinguistic data is viable if all of the data are already associated with textual descriptions such as commentaries or tags. Such textual descriptions are, however, often not available nor suitable to be matched with the queries directly.

To address this problem, we propose a novel approach for building a retrieval system for nonlinguistic data that accepts keyword queries. Our method assumes that there are some amount of game records that are annotated with human-generated commentaries. By using such data, we train a neural network-based symbol grounding model that can associate each of the entries in the database with terms (*i.e.* words or phrases) that characterize it.

In this work, we use *Shogi* (Japanese chess) as a test-bed for our proposed approach. This is primarily because some of the game records of professional players are available with commentaries, which allows us to carry out nontrivial experiments. Also, the search functionality for game states in *Shogi* is useful in its own right. It, for example, allows learners to find game records of a specific opening that they want to study or helps professional players write a book about particular defensive configurations (castles) or strategies. Our work saves them from learning system-specific state notations to be matched with game states. Moreover, our method can naturally cover the states that are not annotated with human-generated comments. We can therefore even search the rapidly growing collection of game records made by computer *Shogi* programs.

2 RELATED WORK

There are attempts at generating sentences for images [15, 17]. Those methods allow one to build a keyword-based search system by using the sentences generated automatically. In such studies the natural language annotations for training the model are usually provided by crowdsourcing. By contrast, we use spontaneous commentaries made by experts for game fans. The symbol grounding in our setting

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SIGIR'17, Tokyo, Japan

© 2016 Copyright held by the owner/author(s). 123-4567-24-567/08/06...\$15.00
DOI: 10.475/123.4

is therefore more difficult since the commentaries include statements that are not directly relevant to the game states.

As for game state retrieval, the standard method requires the user to specify the whole (or parts) of the target state as a query. Yokoyama and Hiraga [18] and Viriyayudhakorn et al. [16] proposed such systems for *Shogi*. Their systems are useful if the users can specify the states they want. Gan-guly et al. [1] have proposed a more sophisticated state-based search method for chess. Their method, taking chess rules into account, enables users to search for similar chess states. In contrast to their system, we train a symbol grounding model to associate game states with their characteristic terms. With our method, users can search for typical piece formations including variations of a strategy or a castle even if they do not remember the precise formation of the pieces.

3 GAME AND COMMENTARY

Shogi is a two-player board game similar to chess, where the goal of each player is to capture the opponent’s king. In addition to six kinds of chess-like pieces with similar moves, *Shogi* has three more kinds of pieces: gold, silver, and lance. The biggest difference from chess is that a player can place a captured piece back on the board as one of his own.

High-profile *Shogi* matches are usually broadcast with commentaries made by experts, who explain the reasoning behind moves, evaluate the current state, and suggest probable moves. These commentaries are made available with the corresponding game states, which allowed us to obtain many pairs of game states and commentary sentences [6].

Commentaries contain many domain-specific expressions (words or multi-word expressions), most of which are named entities (NEs) [10, 12]. *Shogi* players are familiar with these expressions (*e.g.* King’s gambit and Ruy Lopez in the chess case) and category names (*e.g.* opening).

To facilitate symbol grounding, we have adopted the publicly available game commentary corpus [6]. In the corpus, words denoting a strategy (St) or a castle (Ca) in 1,777 sentences are annotated with BIO tags similar to the general NE [8]. We call them *Shogi* NEs (s-NE for short).

4 PROPOSED METHOD

The key idea of our approach is to build a model to associate new (*i.e.* unknown) states with s-NEs and their probabilities.

4.1 Preprocessing of Commentaries

Before symbol grounding, we segment commentaries into words by a publicly available tool [7], which we adapted to the *Shogi* domain by the *Shogi* corpus [6]¹. Then we recognize s-NEs by an automatic s-NE recognizer [9], which we trained by the *Shogi* corpus. Its accuracy is about 90%. This results in pairs of a game state and a set of s-NEs.

¹Japanese does not have clear word boundaries.

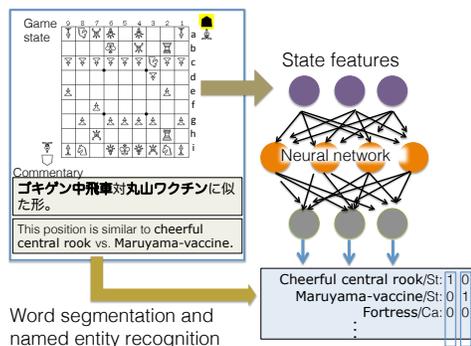


Figure 1: Translating states and commentaries into vectors. This commentary has two s-NEs, then two one-hot vectors are generated.

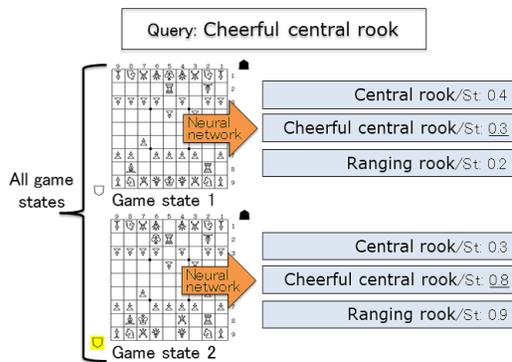


Figure 2: A search example with a query “Cheerful central rook.” State 2 is more relevant than state 1.

4.2 Training the Symbol Grounding Module

Now we have pairs of a game state and a set of s-NEs. Then we make a one-hot vector for each s-NE in the set. Figure 1 shows two one-hot vectors in the bottom right box. Let \mathbf{y}_n be an one-hot vector. Next, we translate a state associated with \mathbf{y}_n into a state feature vector \mathbf{x}_n . As the features we adopt those used in an existing computer *Shogi* program [14]. These features include the values of pieces and the positional relationships of two pieces. Some are real-valued and the others are real-valued. The three units on the top right of Figure 1 correspond to each element of the vector.

Now we have many pairs of \mathbf{y}_n and \mathbf{x}_n to train the symbol grounding module, which is based on a multi-layer feed-forward network. The loss function is the cross-entropy (CE) defined as follows:

$$CE = - \sum_{n=1}^N \sum_{k=1}^K y_{n,k} \log f_k(\mathbf{x}_n),$$

where K is the dimension of the one-hot vector \mathbf{y}_n , and N is the batch size of the training data. We use the softmax function for the output layer. When \mathbf{x}_n is input, $f_k(\mathbf{x}_n)$

means k -th s-NE’s probability of the output and $y_{n,k}$ means whether k -th s-NE is in the set or not.

4.3 Searching

Once we have built the symbol grounding module based on a DNN, we use it to enumerate s-NEs, *i.e.* characteristic terms, with their probabilities associated to the given new state. The probability of the i -th characteristic term (i -th s-NE) is $s_{n,i} = f_i(\mathbf{x}_n)$. We perform the above process for all the states in the game record database.

Now we have a database of game states and their probability vector \mathbf{s}_n as shown in Figure 2. Given a characteristic term as the search query to be matched with the i -th s-NE, our method enumerates the states in descending order of $s_{n,i}$. For example, as Figure 2 shows, given “Cheerful central rook” as the query, our method returns the states in descending order of the underlined probabilities.

5 EXPERIMENTAL EVALUATION

5.1 Experimental Settings

For DNN training we extracted pairs of a state and a commentary from records of real professional matches. We ran our s-NE recognizer and obtained 314 s-NE types. We filter commentaries with the condition that they include one or more s-NEs. As a result we obtained 4,645 pairs of a state and the one-hot vector for training.

Our DNN consists of 5 layers. The dimension of a state feature vector \mathbf{x}_n is 136,045, and each hidden layer has 300 units. On the third layer, dropout [11] is applied (ratio = 0.5). The dimension of the vector \mathbf{y}_n is 314, which is equal to the number of s-NE types found in the corpus. We trained the DNN for 100 epochs with the batch size set to 100. We used Adam [5] for optimization. As the activate function, we have adopted ReLU [2].

5.2 Experiment and Result

5.2.1 First Experiment. As the state database for the first retrieval test, we prepared 997 matches consisting of 110,962 states, and associated them with characteristic terms by the symbol grounding module. As search queries we selected the five most frequent characteristic terms in the training commentaries for both castles and strategies.

We retrieved the 20 most relevant states for each query by our method and evaluated them with the *Shogi* dataset [6]. This dataset includes some annotations on relation evaluation between a state and s-NE. The evaluations were conducted according to the following choices by 3 *Shogi* fans.

- Y: Yes, the state is called so.
- F: Not yet. But a player is clearly aiming at it.
- N: No, the state is not called so.
- M: It is impossible to decide.

Thus the number of evaluations is 600 (= (5 + 5) × 20 × 3).

We use a full-text search method as the baseline. 33.1% (= 36,783/110,962) of the states for the tests have human-generated commentaries. We search for the term of each query in the full text of these commentaries, and select 20 states sampled at random among them. Note that the baseline method can only retrieve states with commentaries, while our method does not suffer from such a limitation.

Table 1: Relevance of the retrieved states to the queries.

Category	Query	Proposed method				Full-text search			
		Y	F	N	M	Y	F	N	M
Castle	Bear-in-the-hole	30	3	17	10	18	9	6	27
	Fortress	27	11	15	7	11	16	15	18
	Silver crown	13	2	38	7	18	12	24	6
	Cape castle	60	0	0	0	22	14	17	7
	Left-cape castle	23	17	3	17	8	9	21	22
Subtotal		153	33	73	41	77	60	83	80
		62.0%		38.0%		45.7%		54.3%	
Strategy	Ranging rook	58	0	1	1	41	2	10	7
	Static rook	43	2	10	5	50	0	0	10
	Central rook	54	5	0	1	27	4	14	15
	Fourth-file rook	58	0	2	0	42	0	14	4
	Climbing silver	38	0	21	1	29	8	9	14
Subtotal		251	7	34	8	189	14	47	50
		86.0%		14.0%		67.7%		32.2%	
Total		404	40	107	49	266	74	130	130
		74.0%		26.0%		56.7%		43.3%	

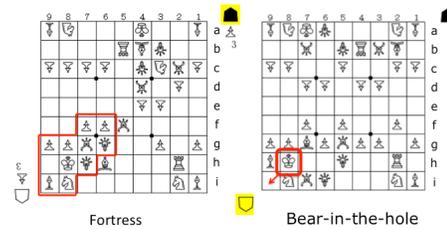


Figure 3: Positive examples. These positions represent the strategies, “Fortress” and “Bear-in-the-hole.”

5.2.2 First Experimental Result. Because users usually want to retrieve states that can be or will be matched with the query, we regard Y and F as correct. The others are regarded as incorrect. The results are shown in Table 1.

We see that 74.0% of the states retrieved by the proposed method are correct, which is much higher than 56.7% of those of full-text search. One reason for the low performance of the baseline is that the query does not always match the meaning of the sentence. An example is “The player should **not** select Cape castle” (evaluation: N, M, N). The state associated with this sentence should not become a position of “Cape castle.” Our method, on the other hand, does not suffer from such fake relations because they are not so frequent in the data. In addition, our method performs better than the baseline in terms of the coverage, because the baseline method cannot search for the states without commentaries. These are the reasons why our method performs better.

Figure 3 shows results of the proposed method. The state on the left side is the typical castle of “fortress” (evaluation: Y, Y, Y). The positions enclosed in bold lines characterize the “fortress.” The state on the right side is a state before the construction of the “bear-in-the-hole” castle. The bear (king) at 8h has not entered in the hole (9i) yet, but Black is clearly aiming to make that castle (evaluation: F, F, Y).

5.2.3 Second Experiment. We also conducted a retrieval experiment with a short-sentence query. Our method needs keywords as queries; thus, the sentences, corresponding to these queries, are converted into keywords by word segmentation and s-NE recognition. When the sentence has one or more s-NEs, we define that the score of the state is the

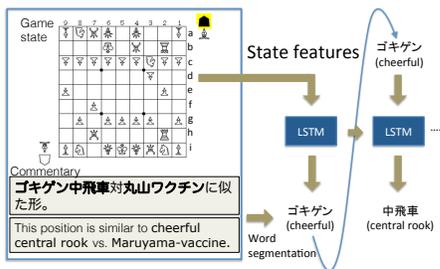


Figure 4: Training of the LSTM. This model aims to reproduce an original commentary from the state features with conditional probability.

Table 2: Comparison of the proposed method and baseline in the second experiment.

	Proposed method	LSTM-model	Chance rate
MRR	0.13944	0.07155	0.02124

product of the output probabilities of all s-NE queries, and rank states according to the scores.

We prepared 295 ($=M$) states with corresponding commentaries. We try to find the prepared state by using the commentary as a query. Naturally, when searching for states, we do not use correspondence information between the commentary and the state. As the evaluation method, we use the mean reciprocal rank $MRR = \frac{1}{M} \sum_{m=1}^M \frac{1}{r_m}$, where r_m is the rank of the corresponding state of the m -th query. We conducted experiments on all sentences and calculated it.

We train an LSTM language model [3], which is widely used for text generation from images, as a baseline method with the same dataset as the proposed method (Figure 4). An LSTM can map from a feature vector to the associated commentary with its probability, and rank the states with the conditional probability of the sentence-query given an input state feature vector. The hidden layer has 300 units. The dimension of the output layer is 7,776, which is equal to the number of word types in the corpus. Other settings are the same as the proposed method.

5.2.4 Second Experimental Result. Table 2 shows the result of the second experiment. The size of the test data is 295 ($=M$). The chance rate of MRR, which selects a state at random, is $0.02124 = \frac{1}{295} \sum_{m=1}^{295} \frac{1}{m}$. This result shows that the proposed method is superior to the baseline LSTM model and chance rate.

These results demonstrate our system enables the game state search by keywords queries with our symbol grounding module, which successfully captures characteristics of strategies or castles.

6 CONCLUSION

In this paper, we have presented a search method for nonlinguistic data with keyword queries, taking game states as a concrete example. Our method first associates game states with characteristic terms by a DNN-based symbol grounding module trained on a small number of pairs of a state and

human-generated commentaries. Given keyword queries, our method enumerates the states that are associated with the queries. Thus our method can retrieve states even if they do not have any commentaries. The experimental results show that our method considerably outperforms the text-based and a simple LSTM model.

Our method uses a domain-specific feature design and a term recognizer. Future work includes the application of this framework to other nonlinguistic data, such as stock charts and medical data.

REFERENCES

- [1] Debasis Ganguly, Johannes Leveling, and Gareth JF Jones. 2014. Retrieval of similar chess positions. In *Proc. of the SIGIR14*. ACM, 687–696.
- [2] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *Proc. of the AISTATS15*, Vol. 15. 275.
- [3] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
- [4] Jiwoon Jeon, Victor Lavrenko, and Raghavan Manmatha. 2003. Automatic image annotation and retrieval using cross-media relevance models. In *Proc. of the SIGIR03*. ACM, 119–126.
- [5] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [6] Shinsuke Mori, John Richardson, Atsushi Ushiku, Tetsuro Sasada, Hirotaka Kameko, and Yoshimasa Tsuruoka. 2016. A Japanese Chess Commentary Corpus. In *Proc. of the LREC16*.
- [7] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. 2011. Pointwise Prediction for Robust, Adaptable Japanese Morphological Analysis. In *Proc. of the ACL11*. 529–533.
- [8] Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proc of the CoNLL03*. 142–147.
- [9] Tetsuro Sasada, Shinsuke Mori, Tatsuya Kawahara, and Yoko Yamakata. 2015. Named Entity Recognizer Trainable from Partially Annotated Data. In *Proc of the PACLING15*.
- [10] Burr Settles. 2004. Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*. 33–38.
- [11] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.
- [12] Yuka Tateisi, Jin-Dong Kim, and Tomoko Ohta. 2002. The GENIA Corpus: an Annotated Research Abstract Corpus in Molecular Biology Domain. In *Proc. of the HLT02*. 73–77.
- [13] Sudeep D Thepade and Ashvini A Tonge. 2014. An optimized key frame extraction for detection of near duplicates in content based video retrieval. In *Proc. of the ICCSP14*. IEEE, 1087–1091.
- [14] Yoshimasa Tsuruoka, Daisaku Yokoyama, and Takashi Chikayama. 2002. Game-tree Search Algorithm based on Realization Probability. *ICGA Journal* 25, 3 (2002), 145–152.
- [15] Yoshitaka Ushiku, Tatsuya Harada, and Yasuo Kuniyoshi. 2011. Automatic Sentence Generation from Images. In *Proc. of the ACM MM11*. 1533–1536.
- [16] Kobkrit Viriyayudhakorn and Mizuhito Ogawa. 2011. Associative Search on Shogi Game Records. *Information and Media Technologies* 6, 3 (2011), 833–845.
- [17] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proc of the ICML15*, Vol. 14. 77–81.
- [18] Hiroshi Yokoyama and Yuzuru Hiraga. 2005. A Shogi Score Retrieval System based on Inverted Index. *IPSJ SIG Technical Reports. GI* 2005, 17 (2005), 19–26.