

# Estimation of Interpersonal Relationships in Movies

Yuta OHWATARI<sup>†a)</sup>, Takahiro KAWAMURA<sup>†b)</sup>, Nonmembers, Yuichi SEI<sup>†c)</sup>, Member,  
Yasuyuki TAHARA<sup>†d)</sup>, Nonmember, and Akihiko OHSUGA<sup>†e)</sup>, Member

**SUMMARY** In many movies, social conditions and awareness of the issues of the times are depicted in any form. Even if fantasy and science fiction are works far from reality, the character relationship does mirror the real world. Therefore, we try to understand social conditions of the real world by analyzing the movie. As a way to analyze the movies, we propose a method of estimating interpersonal relationships of the characters, using a machine learning technique called Markov Logic Network (MLN) from movie script databases on the Web. The MLN is a probabilistic logic network that can describe the relationships between characters, which are not necessarily satisfied on every line. In experiments, we confirmed that our proposed method can estimate favors between the characters in a movie with F-measure of 58.7%. Finally, by comparing the relationships with social indicators, we discussed the relevance of the movies to the real world.  
**key words:** Markov Logic Network, semantic analysis, Open Movie Database, perception/cognitive metrics

## 1. Introduction

Many movies have been filmed in the last couple of decades. In the majority of them, social conditions and awareness of certain issues are constantly depicted in the era of their production. Therefore, use of the movies for the social understanding requires a quantitative metrics. We thus focused on the character relationship in movies in this paper, since such an interpersonal relationship is a mirror of the real world in many movies, even in fantasy and sci-fi stories. For example, in the “X-Men” movie series, conflicts between mutants and humans are often compared to conflicts in the real world that are experienced by minority groups in the US, such as African Americans, Jews, atheists, Communists, and the LGBT community. “Magneto” and “Professor X” in the X-men represent “Malcolm X” and “Martin Luther King” in the African-American Civil Rights Movement. Therefore, we assumed that the character relationship in the movies can be useful metrics for understanding the contemporary social conditions and issues. In fact, there are teaching courses of studying movies for Managing Cross-Cultural Issues in MBA students [1] and online interactive exercises [2].

Manuscript received March 30, 2015.

Manuscript revised August 8, 2015.

Manuscript publicized November 5, 2015.

<sup>†</sup>The authors are with The University of Electro-Communications, Chofu-shi, 182–8585 Japan.

a) E-mail: y-ohwatari@ohsuga.is.uec.ac.jp

b) E-mail: kawamura@ohsuga.is.uec.ac.jp

c) E-mail: sei@is.uec.ac.jp

d) E-mail: tahara@is.uec.ac.jp

e) E-mail: ohsuga@uec.ac.jp

DOI: 10.1587/transinf.2015MUP0015

We first estimate a character diagram, which is a graph representing the relationships of the characters in a movie. Movies are produced based on scripts, thus we used the scripts for estimating by a machine learning technique, called Markov Logic Network (MLN). Then, finally we try to compare social indicators such as divorce rates and the stability of the character relationships in the movies.

In this paper, we begin by briefly reviewing the necessary background of the MLN in Sect. 2. We then describe our proposed method to estimate the interpersonal relationship between characters in a movie in Sect. 3, and report on our experiments in Sect. 4, and report classification of estimated relationships and discuss the relevance of the connection between the real world and the movies in Sect. 5. Finally, we describe related work, and outline directions for future work in Sect. 6 and Sect. 7.

## 2. Background

In this section, we introduce the Markov Logic Network. First, we briefly review the first-order predicate logic and the Markov Network. Then, we describe the Markov Logic Network.

### 2.1 First-Order Predicate Logic

A first-order knowledge base is a collection of texts and rules in first-order logic [3]. Rules are constructed using 4 types of symbols that are constants, variables, functions, and predicates. In a region, constants are the objects and variables are scope over the objects. Functions are mappings from a set of objects to another object. Predicates are connections between the objects and other objects. Table 1 shows examples of these symbols.

A ground term is an representation of objects not using variables in a region. Then, a ground predicate is an atomic formula by using only arguments are ground terms. Using logical connectives and quantifiers, formulas are recursively constructed. A knowledge base in a clausal form is a con-

**Table 1** Example of symbols

Symbol	Example
Constant	$people = \{Anna, Bob, Chris, \dots\}$
Variable	$x$
Function	$MotherOf(x)$
Predicate	$Friends(x, y), Smokes(x)$

junction of clauses that is a disjunction of literals. When each possible ground atom is assigned a truth value, we call such a situation with the possible world. By replacing each quantified formula with a conjunction or disjunction of all its groundings in finite regions, first-order knowledge base enable to be propositionalized.

A problem in logic is about determining if a knowledge base is satisfiable. So, a model checking is nothing more but satisfiability testing. The main approaches in model checking are backtracking (e.g. DPLL) and stochastic local search (e.g. WalkSAT) [4], [5].

A first-order knowledge base can be seen as a set of hard constraints on the set of possible worlds. However, the solution in the real world is often placed on a set of impossible worlds. In contrast, the MLN solves this problem by associating a weight that reflects how strong a constraint with each formula. The Markov Logic Network enables us to estimate while also violating the formula to a certain degree.

## 2.2 Markov Network

A Markov Network is a model for the joint distribution of a collection of variables and a collection of potential functions  $\phi_k$  [6] and, is composed of an undirected graph  $G$ . This is also known as a Markov Random Field. The graph  $G$  has a node for each variable. Furthermore, this model has a potential function for each clique in the graph  $G$ . If a distribution is strictly positive, and a graph encodes conditional independence, the distribution is the product of potential functions over cliques in the graph.

The joint distribution represented by a Markov Network is as follows:

$$P(X = x) = \frac{1}{Z} \prod_k \phi_k(x_{[k]}) \quad (1)$$

$$= \frac{1}{Z} \exp\left(\sum_j w_j f_j(x)\right) \quad (2)$$

Where  $x_{[k]}$  is the state of the  $k$ th clique.  $Z$  is known as the partition function, and it is given by  $Z = \sum_{x \in X} \prod_k \phi_k(x_{[k]})$ . Equation (2) is represented as the log-linear model of Eq. (1), with each clique potential replaced by an exponentiated weighted sum of features of the state. We will focus on binary real-valued function of the state,  $f_j(x) \in \{0, 1\}$ . Its weight being  $\log \phi_k(x_{[k]})$  represents each possible state  $x_{[k]}$  of each clique.

Approaches of inference in Markov Network are computing probabilities (e.g. MCMC, Belief Propagation) [7], [8] and MAP inference [9]. On the other hand, it is laborious to construct Markov Network. However, constructing Markov Network from Markov Logic Network is straightforward, since the first-order predicate logic represents an initial template of Markov Network.

## 2.3 Markov Logic Network

The MLN is a probabilistic extension of finite first-order logic that makes up the disadvantages of all the other Markov Network and first-order logic [10]. The MLN is a set of pairs formula  $F_i$  in first-order logic and its weight  $w_i$ . The following equation is the probability distribution over possible world  $x$  specified by the ground MLN.

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_i w_i n_i(x)\right) \quad (3)$$

$Z$  is a normalization term and  $n_i(x)$  is the number of true groundings of  $F_i$  in  $x$ . The exponent in Eq. (3) is the sum of the weights of the satisfied ground clauses, and thus  $P(X = x)$  can be optimized by maximizing this sum. The MAP inference in the MLN can be carried out efficiently using a weighted satisfiability solver like MaxWalkSAT [5]. Using a set of formulas, their weights can be learned by either maximizing the conditional likelihood of the query predicates or maximizing the joint likelihood of all predicates [11], [12].

The MLN has been used for the joint inference on the sentiment polarity of sentences and documents [13], entity resolution [14], information extraction [15], predicate-structure analysis of Japanese [16], and inference on temporal relation identification [17]. In these studies, They adopted the MLN because of requiring fuzzy rules in addition to the strict rules. For similar reasons, we also adopted MLN for representing the interpersonal relationships and reflecting the tendency.

Note that we used the learning and inference algorithms provided in the open-source Alchemy 2.0<sup>†</sup> package as the implementation of the MLN in this paper.

Bayesian Network is a stochastic approach similar to Markov Network. The advantages of MLN are the first-order predicate logic, which can be a template language for constructing the networks and various algorithms for inference and learning in the Markov Network.

## 3. Proposed Method

An overview of our proposed method is illustrated in Fig. 1. Our method can be separated into the estimation of interpersonal relationships and the generation of character diagrams.

First, we prepared learning and inference data (speakers, listeners, and contents), by extracting who speak what, and to whom from a movie script. Then, we estimate the sentiment polarity for lines in the script and the favorable impression between a speaker and a listener using MLN. Finally, we generate the character diagram of a movie from estimated interpersonal relationships.

We show a part of data for learning and inference in Table 2. *Likes* rules are prepared only for the training data. That is, we determine the probabilities of *Likes* from the

<sup>†</sup><http://alchemy.cs.washington.edu/>

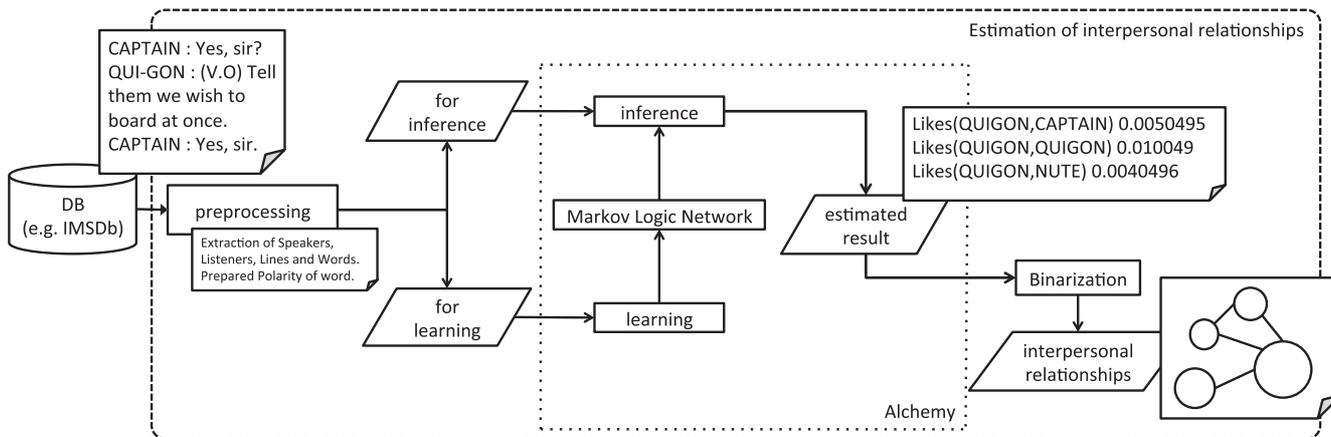


Fig. 1 Method overview

Table 2 Example of movie script data

Notation	Description
Line("Yes, sir?", CAPTAIN, QUIGON)	CAPTAIN speak "Yes, sir?" to QUIGON
Word("Yes, sir?", 2, "sir")	"sir" is the 2nd position in "Yes, sir?"
Likes(ANAKIN, PADME)	ANAKIN likes PADME

training data, and estimate the probabilities of *Likes* in the test data.

In the remainder of this section, we describe logical expressions and rules used in our proposed method. We then describe about how to determine the favorability of two characters from the estimated results.

### 3.1 Defined Rules

We describe the definitions of the rules to estimate interpersonal relationships from the sentiment of line. In MLN, these formulas function as a weak constraint by learning. This, we expect as advantages of using the MLN, the MLN enables to grasp ambiguity that he/she actually likes while person referred to as dislike. The sentiment polarity of a word is a defined attribute and each word has either a positive or negative polarity. For example, a word such as "good" or "beautiful" has a positive polarity while a word such as "bad" or "dirty" has a negative polarity. As the reference of sentiment polarity words, we used the Semantic Orientations of Words Dictionary that is built by Takamura et al. [18]. The dictionary assigned a real value in the range  $-1$  to  $+1$  to each word, where the values close to  $-1$  are supposed to be negative, and the values close to  $+1$  are positive. The vocabulary was extracted from WordNet<sup>†</sup>. In this paper, we assigned positive or negative polarity, depending on whether the value of semantic polarity of words in the dictionary is more than 0 or not. The sentiment polarity of lines in movie scripts also has a positive or negative polarity, which is described in the next section. Arguments of observed predicates are given in both inference and learning phase, but the arguments of hidden predicates are not given in in-

<sup>†</sup><http://wordnet.princeton.edu>

Table 3 Observed predicate

Predicate	Description
Line(text, speaker, listener)	<i>speaker</i> speak <i>text</i> to <i>listener</i>
Word(text, position, word)	<i>word</i> in <i>text</i> and the position is <i>position</i>
Wpol(word, polarity)	The sentiment polarity of <i>word</i> is <i>polarity</i>

Table 4 Hidden predicate

Predicate	Description
Lpol(text, polarity)	The sentiment polarity of <i>text</i> is <i>polarity</i>
Likes(person1, person2)	Favor
Mutual(person1, person2)	Both feelings of favor
Unreturned(person1, person2)	Unrequited love
Oppose(person1, person2)	Opposition

ference phase. Observed predicates and hidden predicates in this paper are shown in Table 3 and Table 4. For example, *Line(text, speaker, listener)* shows that the *speaker* speaks to the *listener* with *text*, and *Lpol(text, polarity)* shows that a sentiment polarity of *text* is *polarity*.

#### 3.1.1 Estimation of Sentiment Polarity of Lines

We describe logical formulas for movie lines in this section. As mentioned above, *text* and *position* in formulas are variables, and corresponding data are assigned from movie scripts. Constants are enclosed in double quotes. If  $+$  gets attached to the beginning of the variable, it is replaced by a constant that will be replaced by the actual data (grounding) at the execution time. For example, the rule *Lpol(text, +polarity)* is replaced in each *Lpol(text, "P")*, and *Lpol(text, "N")*.

#### Formula of determining the sentiment polarity of a line

If there is a word (*word*) with a polarity (*polarity*) at any position (*position*) in a line (*text*), the polarity (*polarity*) of the line is the polarity of the word. However, as the following formula can take multiple variables for a line, that is, *polarity* and *position*, the nodes represented by this formula are multiplexed and the probabilities of polarity are calculated for each assignment of *position* in a line. (E.g. the 1st word of the *text A*, the 2nd word of the *text A*, ..., the 1st word of

the text  $B$ , ...)

$$\begin{aligned} & \text{Word}(\text{text}, \text{position}, +\text{word}) \\ & \wedge \text{Wpol}(+\text{word}, +\text{polarity}) \\ & \Rightarrow \text{Lpol}(\text{text}, +\text{polarity}) \end{aligned} \quad (4)$$

### 3.1.2 Estimation of Interpersonal Relationships

In this section, we describe the formulas for estimating the relationships between a listener and a speaker.

#### Formulas of determining a feeling toward the other

These formulas (Eq. (5) and (6)) indicate that favorability of the speaker to the listener is determined by the positive or negative polarity of lines.

$$\begin{aligned} & \text{Line}(\text{text}, +\text{speaker}, +\text{listener}) \\ & \wedge \text{Lpol}(\text{text}, "P") \\ & \Rightarrow \text{Likes}(+\text{speaker}, +\text{listener}) \end{aligned} \quad (5)$$

$$\begin{aligned} & \text{Line}(\text{text}, +\text{speaker}, +\text{listener}) \\ & \wedge \text{Lpol}(\text{text}, "N") \\ & \Rightarrow \neg \text{Likes}(+\text{speaker}, +\text{listener}) \end{aligned} \quad (6)$$

#### Formula of determining to be both friendly

This logical expression (Eq. (7)) shows a relationship that favors the other party in relation to the other. The formula features a relationship that suggests a positive attitude that the participants show towards each other.

$$\begin{aligned} & \text{Likes}(+\text{person1}, +\text{person2}) \\ & \wedge \text{Likes}(+\text{person2}, +\text{person1}) \\ & \Rightarrow \text{Mutual}(+\text{person1}, +\text{person2}) \\ & \wedge \text{Mutual}(+\text{person2}, +\text{person1}) \end{aligned} \quad (7)$$

#### Formula of determining to be friendly unilaterally

The formula (Eq. (8)) shows a relationship that one has a favor for the other, but the other dislikes the opponent.

$$\begin{aligned} & \text{Likes}(+\text{person1}, +\text{person2}) \\ & \wedge \neg \text{Likes}(+\text{person2}, +\text{person1}) \\ & \Rightarrow \text{Unreturned}(+\text{person1}, +\text{person2}) \end{aligned} \quad (8)$$

#### Formula of determining to hate each other

This formula (Eq. (9)) shows the relationship that the participants hate each other.

$$\begin{aligned} & \neg \text{Likes}(+\text{person1}, +\text{person2}) \\ & \wedge \neg \text{Likes}(+\text{person2}, +\text{person1}) \\ & \Rightarrow \text{Oppose}(+\text{person1}, +\text{person2}) \\ & \wedge \text{Oppose}(+\text{person2}, +\text{person1}) \end{aligned} \quad (9)$$

**Table 5** Information of movie scripts in experiment

#	Title	Released year
1	Jaws	1975
2	Rocky	1976
3	Star Wars Episode IV A New Hope	1977
4	Jaws2	1978
5	Kramer vs. Kramer	1979
6	Star Wars Episode V The Empire Strikes Back	1980
7	Raiders of the Lost Ark	1981
8	Star Trek II: The Wrath of Khan	1982
9	Star Wars Episode VI Return of the Jedi	1983
10	Ghostbusters	1984
11	Back to the Future	1985
12	Top Gun	1986
13	Three Men and a Baby	1987
14	Who Framed Roger Rabbit	1988
15	Batman	1989
16	Ghost	1990
17	Terminator 2: Judgment Day	1991
18	Aladdin	1992
19	Jurassic Park	1993
20	Forrest Gump	1994
21	Toy Story	1995
22	Independence Day	1996
23	Titanic	1997
24	Saving Private Ryan	1998
25	Star Wars Episode I The Phantom Menace	1999

### 3.2 Estimation of Character Relationships

The estimation results are expressed as a *Likes* predicate and its weight on each relationships between two characters. Thus, to determine whether there is a favor between two characters, we calculated the average value of the weights of the *Likes* predicates to other characters as a threshold. If the weight is over the threshold, we determined that there is a favor between those two characters.

## 4. Experiment

We conducted experiments to evaluate the estimation accuracy of interpersonal relationships by the proposed method. In this section, we describe the results and discussion.

### 4.1 Datasets

In the experiment, we used movie script data from IMSDb: The Internet Movie Script Database<sup>†</sup> on the Web. Table 5 shows the title and released year of 25 movies used in the experiment.

Table 6 shows the number of lines, persons, relationships of likes, and relationships of dislikes. Then, Table 7 shows the number of ground clauses and ground formulas in the learning phase.

### 4.2 Experimental Setting

We prepared a set of movie scripts in Table 5 for estimation,

<sup>†</sup><http://www.imsdb.com/>

**Table 6** Information of movies and relationships

#	Lines	Persons	Likes	Dislikes
mean	581.08	8.96	33.96	6.72

**Table 7** The number of ground formula and ground clause

#	ground formulas	ground clauses
mean	7619.28	24125.5

and then every movie is as test data, and 5 movies are randomly chosen from the remaining 24 movies as the training data. We tried to evaluate the accuracy not only of *Likes* rules, but also of *Dislike* rules. That is, we calculated indicators (Precision / Recall / F-Measure) including *Dislikes* relationships. However, since the number of *Dislikes* rules is very little in comparison with *Likes* rules, we could not obtain the reliable results of *Dislikes* rules only. So, we evaluated the accuracy of *Likes* and *Dislikes* rules in the estimated interpersonal relationships of the characters, who had a conversation more than three times. The opportunities of application of rules other than *Likes* and *Dislikes* rules were very few. Thus, when we used Mutual, Unreturned, and/or Opposed in a formula, the calculation amount increased but the effect was limited. For example, when we added just a predicate of Mutual, Unreturned, and Opposed, the computation time increased about five times. However, the addition of the predicate did not significantly contribute to the accuracy in the experiment. Then, we compared the accuracy with a method for randomly determining the favors (Random in Table 12). In addition to that, we also calculated the summarized polarity of positive (+1) and negative (-1) for each word. That is, the method is determining the favorability by the frequency of the positive and negative polarities of the words in all words by using for each speaker to each listener (Frequency in Table 12). And, we adopt a typical sentiment classification method using SVM, in reference to B. Pang et al. [19]. In this method, feature vectors are created from the word frequency in the sentiment dictionary based on the words spoken from a speaker to a listener. Then, we labeled *Likes* or *Dislikes* to each vector based on the correct data. Therefore, the prepared SVM method is the classification of the labels (*Likes* or *Dislikes*) from the feature vectors. Note, that we asked two persons for annotation of *Likes* rules in the training data (correct data), who are familiar with the movies and have seen the movies in Table 5 before and the data was confirmed by 5 people.

### 4.3 Results

The experimental results are shown in Table 12. The time required for the training was 22 hours 56 minutes 25.29 seconds, and the time for estimation was 6 minutes 48.08 seconds. The server used for the experiment has 2 CPUs that is Intel(R) Xeon(R) CPU E5-2407 @ 2.20GHz with 32GB memory.

In the results, our proposed method has higher performance than the two baseline methods (randomly and fre-

**Table 8** The confusion matrix of proposed method

		Actual	
		Positive	Negative
Prediction	Positive	33.12	13.84
	Negative	34.44	14.72

**Table 9** The confusion matrix of randomly method

		Actual	
		Positive	Negative
Prediction	Positive	9.88	5.2
	Negative	45.24	18.84

**Table 10** The confusion matrix of frequency method

		Actual	
		Positive	Negative
Prediction	Positive	26.12	10.76
	Negative	41.44	17.56

**Table 11** The confusion matrix of SVM method

		Actual	
		Positive	Negative
Prediction	Positive	154	63
	Negative	305	128

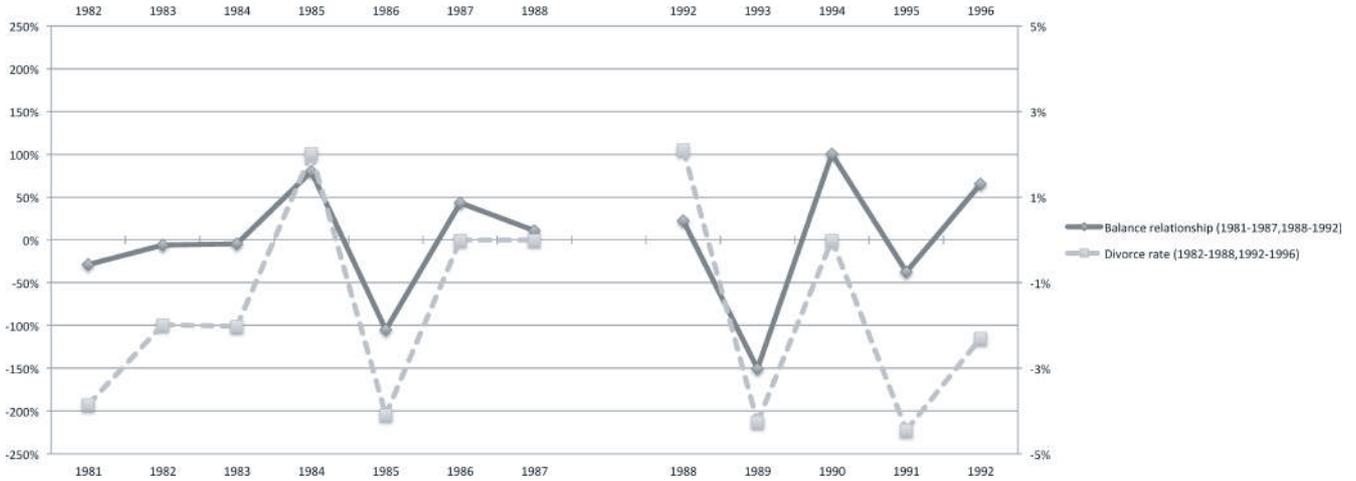
**Table 12** The performance of estimating relationships

#	Precision			
	Proposed	Random	Frequency	SVM
mean	50.5%	37.5%	45.3%	58.8%
#	Recall			
	Proposed	Random	Frequency	SVM
mean	50.0%	19.8%	38.6%	43.4%
#	F-Measure			
	Proposed	Random	Frequency	SVM
mean	58.7%	31.0%	49.9%	44.2%

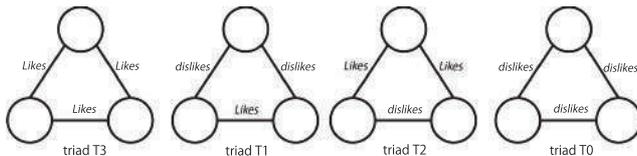
quency) for all and the SVM method except for precision. Since SVM can learn explicit patterns of Likes, the precision gets higher, whereas the recall remains low. As a result, our method obtained higher F-measure. Since our method really closes to the two baseline methods in the precision, we can say that the proposed method is useful compared to the two baseline methods. Furthermore, since the proposed method had better results than the baseline methods, in which frequency of favor and/or hostility words are directly used as indicators of estimation, we can find that the probability inference effectively worked for the estimation.

The confusion matrix of each method are shows in Table 8, Table 9, Table 10, and Table 11. Compared to the two baseline methods, False Positive in our method increased, and decrease of precision was suppressed by significant improvements in True Positive. Since we confirmed that the baseline method using a semantic polarity of words also had the significant improvement of True Positive, the use of semantic polarity of words was effective for estimation of interpersonal relationships. Also, recall of our proposed method was improved by the decrease of False Negative, compared with the baseline method. Since False Negative in





**Fig. 5** Transition of balance relationships rate (1981-1987,1988-1992) and divorce rate in U.S.(1982-1988,1992-1996)



**Fig. 4** Four pattern in undirected interpersonal relationships

ple are friends and dislike the one that remains. T2 is in a state where two people dislike one of the remaining, but the one likes the other two people. T0 is in a state where three people dislike one another.

According to the verification on the social media (E.g. Epinions, Slashdot, and Wikipedia) by Jure Leskovec et al. [23], the number of the relationships with stable balance was five times the relationships with unstable balance.

## 5.2 Methodology

We then converted the estimated relationships into an undirected graph for the characters, who talked more than three times. First, we prepare all the combinations of three persons from all characters in the movie scripts. Then, we calculate the edge score between two persons by summing up the score of the relationship from person A to person B and the score of the relationship from person B to person A. In this way, we determine a type of the combinations of three persons, and classify them into the four pattern triads in Fig. 3. The Positive edge means the relationship of *Likes*. Figure 3 shows an example of graphs with positive edges and negative edges. This figure was generated from *The Lord of the Rings The Fellowship of the Ring (2001)*.

Then, we classified the relationships into four pattern triads shown in Fig. 4.

## 5.3 Results

Table 13 shows the result of the classification of movies in

**Table 13** Classification of estimated relationships

#	T3	T1	T2	T0	Comb. of Edge	All Comb.
mean	0.96	4.84	4.24	1.8	11.84	86.88

Table 5. The number of balanced triads (T3 and T1) was almost the same number as unbalanced triads (T2 and T0). Thus, the estimated relationships did not follow the result of Jure Leskovec et al. In addition to the difference between the real and fictional world, a reason is that the estimated relationships could not cover all the relationships, since the recall was low in our experiments.

In Fig. 5, we illustrated the transition of the percentage of balanced triads (T3 and T1) and the divorce rate in the United States since 1981. The under horizontal axis refers to the released years of movies, the above horizontal axis refers to the years in the divorce rate, the left vertical axis corresponds to the inflation rate, and the vertical axis of the right is the percentage of the balanced triads. In Fig. 5, we can find that the transition of interpersonal relationships in the movies is somewhat similar to the transition of divorce rate. The correlation coefficient of these transitions calculated was  $r = 0.9143 (P = 0.00394404)$  and  $r = 0.6335 (P = 0.251214593)$ . From this result, the divorce rate was changed in accordance with the change of stable relationship rate. Depending on whether the relationship of the character in the movies is friendly or not, divorce rate increased or decreased in the following years.

The comparison was made in the year of movie release, but movies were shot a few years before the release. In other words, the filmmakers might predict the social conditions in the released year of the movie, and consider the most accepted relationships of the characters. Most of the movies in this paper have been nominated for Academy Awards on the American Academy of Motion Picture Arts and Sciences. So, there is no doubt that many people had sympathy for the movies. Thus, it can be considered to be potentially indicative for understanding the social conditions and issues from

the movies as mentioned in Sect. 1.

## 6. Related Work

In this section, we describe the related works in detail. We first considered an approach of using a video recognition technology to summarize a movie. In audio and video recognition technology, there are studies of “speaker clustering” to classify speeches to each speaker [24], and “speaker indexes” to capture “who spoke, and when” [25]. Speaker clustering is a technique that classifies speech utterances from multiple speakers in broadcast news, and meetings, etc. The most conventional clustering methods is batch processing methods, which require all utterances be present before clustering can be executed. The proposed method by Koshinaka et al. can execute online tasks and exhibit high performance compared to the existing research. Speaker indexing, also known as meeting recognition, is a method of estimation of “who spoke, and when” in particularly important topics. Araki et al. tried to improve speaker indexing by using the features of the available microphone arrays that were strategically placed for meeting. However, we decided to handle a movie script instead of a video recognition in this paper, since we can easily collect movie script data from the Web, and movies are produced based on the movie script. In the future work, we could use the video recognition to incorporate emphasis, modulation and interval in the logic formula.

In studies related to text summarization, Tanaka et al. proposed a method to summarize a novel [26]. They can summarize a novel by using relevant keywords and interpersonal relationships from the sentence structure. It seems to be a good method of summarizing the story. However, it was difficult to use this method for movies, since the movie scripts are written only in the format of scenes and dialogues necessary to play. Tsvetomira et al. also proposed a method of summarizing videos automatically, based on textual cues available in subtitles and movie scripts [27]. They extract features like keywords, main characters names and presence, and then combine them in an importance function to identify the moments most relevant to the story line.

On the other hand, there have been studies on estimation of interpersonal relationships in the real world, for example, by analyzing of e-mails [28], and by using co-occurrence of the name [29], and so forth. Adamic et al. tried link analysis of the Web site for extraction of the community and the relationships. Matsuo et al. tried to develop several new algorithms for social networking and mining, such as those that classify relations into categories, and obtained and utilized person-to-word relations by using Web search engines. However, we could not find studies on estimation of interpersonal relationships between fictional characters.

Finally, we introduce studies using MLN. Yokono et al. [13] estimated concurrently sentiment polarity of sentences and a document in contrast to focus the polarity of specific units such as statements and clauses on many of

the existing studies. Entity resolution by using MLN [14] tried to solve the problem of determining which records in a database, refer to the same entities - also known as data cleaning and preparation- is a crucial and expensive step in the data mining process. Information extraction [15] is about extracting database records from text or semi-structured sources. Poon et al. tried a fully joint approach. His solution consisted mainly of writing the appropriate logical formulas, and required much less engineering than previous ones by using MLN. Predicate-structure analysis of Japanese [16] attempted an analysis using MLN considering complex dependencies between term and predicate. The predicate has turned out to have multiple terms and relationships with other predicates. Joint inference in temporal relation identification involves the prediction of temporal order between events and/or time expressions mentioned in the text as well as the relation between events in a document and the time at which the document was created. Yoshikawa et al. tried by using MLN. These studies focus on global constraints, and are capable to build a model easily by using MLN. Building a model have turned out to be quite easy, because it showed a tendency related to interpersonal relationships by which adopting MLN has been deemed rather effective and useful. However, we could not find studies about estimation of interpersonal relationships using MLN.

## 7. Conclusion

In this paper, applying MLN on the movie script database, we estimated the sentiment polarity of lines and the interpersonal relationships of the characters in a movie. In experiments, we confirmed that our method estimated favors between the characters in a movie with a F-Measure of 58.7%. But the experiments still have limitations on the amount of data. We will use much more movies in the future experiments. In the future, we will improve the model to achieve higher accuracy. For example, incorporating the Social Balance Theory and a deeper and more thorough analysis and introduction of the rule could be useful and effective for estimation. We are also planning an automatic visualization of the interpersonal relationships in the movie. The our proposed method failed on an obviously easy case for humans. For example, the relationship between C3PO and Anakin is an easy case for humans. The better to analyze deeper on such cases independently is our future work.

Also, we classified the estimated relationships, and compared with the transition of the social indicators, and then discussed the relevance to the real world.

We hope that the framework we have established in this paper will help better understanding of social conditions and issues in the near future.

## Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 24300005, 26330081, 26870201.

## References

- [1] S. Pandey, "Using popular movies in teaching cross-cultural management," *European Journal of Training and Development*, vol.36, no.2/3, pp.329–350, 2012.
- [2] M.J. Shawback and N. Terhune, "Online interactive courseware: using movies to promote cultural understanding in a call environment," *ReCALL*, vol.14, no.1, pp.85–95, 2002.
- [3] M.R. Genesereth and N.J. Nilsson, *Logical foundations of artificial intelligence*, Morgan Kaufmann, 1987.
- [4] D.S. Johnson and M.A. Trick, *Cliques, coloring, and satisfiability: second DIMACS implementation challenge*, Oct. 1993, American Mathematical Soc., 1996.
- [5] D. Dingzhu, J. Gu, and P.M. Pardalos, eds., *Satisfiability problem: theory and applications: DIMACS Workshop*, March 1996, American Mathematical Soc., 1997.
- [6] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann, 1988.
- [7] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, "Introducing Markov chain Monte Carlo," in *Markov chain Monte Carlo in practice*, pp.1–19, Springer, 1996.
- [8] D. Roth, "On the hardness of approximate reasoning," *Artificial Intelligence*, vol.82, no.1-2, pp.273–302, 1996.
- [9] S. Della Pietra, V. Della Pietra, and J. Lafferty, "Inducing features of random fields," *Pattern Analysis and Machine Intelligence*, *IEEE Transactions on*, vol.19, no.4, pp.380–393, 1997.
- [10] M. Richardson and P. Domingos, "Markov logic networks," *Machine learning*, vol.62, no.1-2, pp.107–136, 2006.
- [11] P. Singla and P. Domingos, "Discriminative training of Markov logic networks," *AAAI*, pp.868–873, 2005.
- [12] H. Poon and P. Domingos, "Sound and efficient inference with probabilistic and deterministic dependencies," *AAAI*, pp.458–463, 2006.
- [13] H. Yokono and M. Okuno, "Joint estimation of sentiment polarity for text and sentence," *Proc. 16th The Association for Natural Language Processing Annual meeting*, pp.182–185, 2010.
- [14] P. Singla and P. Domingos, "Entity resolution with Markov logic," *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pp.572–582, IEEE, 2006.
- [15] H. Poon and P. Domingos, "Joint inference in information extraction," *AAAI*, pp.913–918, 2007.
- [16] K. Yoshikawa, M. Asahara, and Y. Matsumoto, "Japanese predicate argument structure analysis with Markov logic," *Journal of natural language processing*, vol.20, no.2, pp.251–271, 2013.
- [17] K. Yoshikawa, S. Riedel, M. Asahara, and Y. Matsumoto, "Joint inference of temporal relations identification with Markov logic," *Journal of The Japanese Society for Artificial Intelligence*, vol.24, no.6, pp.521–530, 2009.
- [18] H. Takamura, T. Inui, and M. Okumura, "Extracting semantic orientations of words using spin model," *Proc. the 43rd Annual Meeting on Association for Computational Linguistics*, pp.133–140, Association for Computational Linguistics, 2005.
- [19] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pp.79–86, Association for Computational Linguistics, 2002.
- [20] D. Khanafiah and H. Situngkir, "Social balance theory," *arXiv preprint nlin/0405041*, pp.1–12, 2004.
- [21] F. Heider, "Attitudes and cognitive organization," *The Journal of psychology*, vol.21, no.1, pp.107–112, 1946.
- [22] H.F. Taylor, "Balance and change in the two-person group," *Sociometry*, vol.30, no.3, pp.262–279, 1967.
- [23] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Signed networks in social media," *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '10, New York, NY, USA*, pp.1361–1370, ACM, 2010.
- [24] T. Koshinaka, K. Nagatomo, and K. Shinoda, "Online speaker clustering using incremental learning of an ergodic hidden Markov model," *IEICE TRANSACTIONS on Information and Systems*, vol.E95.D, no.10, pp.2469–2478, 2012.
- [25] S. Araki, M. Fujimoto, K. Ishizuka, H. Sawada, and S. Makino, "Speaker indexing and speech enhancement in real meetings/conversations," *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp.93–96, IEEE, 2008.
- [26] S. Tanaka, M. Okabe, and R. Onai, "Interactive narrative summarization," *Workshop on Interactive Systems and Software*, pp.06–01–06–6, 2011.
- [27] T. Tsoneva, M. Barbieri, and H. Weda, "Automated summarization of narrative video on a semantic level," *Semantic Computing, 2007. ICSC 2007. International Conference on*, pp.169–176, IEEE, 2007.
- [28] L.A. Adamic and E. Adar, "Friends and neighbors on the web," *Social networks*, vol.25, no.3, pp.211–230, 2003.
- [29] Y. Matsuo, H. Tomobe, K. Hashida, H. Nakajima, and M. Ishizuka, "Social network extraction from the web information," *Transactions of the Japanese Society for Artificial Intelligence*, vol.20, no.1, pp.46–56, 2005.



**Yuta Ohwatari** received BE degree in electro-communications, from the University of Electro-Communications (UEC). In 2015, he received ME degree in information systems, from from The University of Electro-Communications (UEC).



**Takahiro Kawamura** is a Senior Research Scientist at Corporate Research & Development Center, Toshiba Corp., and also an Associate Professor in the Graduate School of Information Systems, The University of Electro-Communications, a Lecturer in the Graduate School of Engineering, Osaka University, Japan. From 2001 to 2002 he was a Visiting Scholar in Robotics Institute, Carnegie Mellon University. His current research interests include semantic web, open data, and software agent.



**Yuichi Sei** received his Ph.D. degree in Information Science and Technology, from the University of Tokyo, Japan, in 2009. From 2009 to 2012 he was working at Mitsubishi Research Institute. Since 2013, he has been an assistant professor at The University of Electro-Communications. His research interests include pervasive computing, security, and privacy-preserving data mining.



**Yasuyuki Tahara** is an associate professor in The University of Electro-Communications. He received his BSc and his MSc in Mathematics from the University of Tokyo, Japan, and his Ph.D. in Information and Computer Science from Waseda University, Japan, in 1989, 1991, and 2003, respectively. He joined Toshiba Corporation in 1991. He was a visiting researcher in City University London, UK, from 1995 to 1996, and in Imperial College London, UK, from 1996 to 1997. He left Toshiba Corporation

and joined NII in 2003. He left NII and joined The University of Electro-Communications in 2008. His research interests include formal verification of software and requirements engineering. He has been working for an education program called Top SE since 2004. Prof. Tahara is a member of the Information Processing Society of Japan and Japan Society for Software Science and Technology.



**Akihiko Ohsuga** received a B.S. degree in mathematics from Sophia University in 1981 and a Ph.D. degree in computer science from Waseda University in 1995. From 1981 to 2007 he worked with Toshiba Corporation. Since April 2007, he has been a professor in the Graduate School of Information Systems, The University of Electro-Communications (UEC). Since April 2012, he has been also a visiting professor in National Institute of Informatics (NII). He has published more than 210 papers

(90 papers at journals and 120 papers at international conferences). His research interests include agent technologies, web intelligence, and software engineering. He is a member of the IEEE Computer Society, the Information Processing Society of Japan (IPSJ), the Institute of Electronics, Information and Communication Engineers (IEICE), The Japanese Society for Artificial Intelligence (JSAI), and Japan Society for Software Science and Technology (JSSST). He received the 1986 Paper Award from the Information Processing Society of Japan.