Investigation of annotator's behaviour using eye-tracking data

Ryu Iida **Koh Mitsuda** **Takenobu Tokunaga**
Department of Computer Science, Tokyo Institute of Technology
{ryu-i,mitsudak,take}@cl.cs.titech.ac.jp

Abstract

This paper presents an analysis of an annotator's behaviour during her/his annotation process for eliciting useful information for natural language processing (NLP) tasks. Text annotation is essential for machine learning-based NLP where annotated texts are used for both training and evaluating supervised systems. Since an annotator's behaviour during annotation can be seen as reflecting her/his cognitive process during her/his attempt to understand the text for annotation, analysing the process of text annotation has potential to reveal useful information for NLP tasks, in particular semantic and discourse processing that require deeper language understanding. We conducted an experiment for collecting annotator actions and eye gaze during the annotation of predicate-argument relations in Japanese texts. Our analysis of the collected data suggests that obtained insight into human annotation behaviour is useful for exploring effective linguistic features in machine learning-based approaches.

1 Introduction

Text annotation is essential for machine learning (ML)-based natural language processing (NLP) where annotated texts are used for both training and evaluating supervised systems. This annotation-then-learning approach has been broadly applied to various NLP tasks, ranging from shallow processing tasks, such as POS tagging and NP chunking, to tasks requiring deeper linguistic information, such as coreference resolution and discourse relation classification, and has been largely successful for shallow NLP tasks in particular. The key to this success is how useful information can be effectively introduced into ML algorithms as features. With shallow NLP tasks, surface information like words and their POS within a window of a certain size can be easily employed as useful features. In contrast, in semantic and discourse processing, such as coreference resolution and discourse structure analysis, it is not trivial to employ as features deeper linguistic knowledge and human linguistic intuition that are indispensable for these tasks. In order to improve system performance, past attempts have integrated deeper linguistic knowledge through manually constructed linguistic resources such as WordNet (Miller, 1995) and linguistic theories such as Centering Theory (Grosz et al., 1995). They partially succeed in improving performance, but there is still room for further improvement (duVerle and Prendinger, 2009; Ng, 2010; Lin et al., 2010; Pradhan et al., 2012).

Unlike past attempts relying on heuristic feature engineering, we take a cognitive science approach to improving system performance. In stead of employing existing resources and theories, we look into human behaviour during annotation and elicit useful information for NLP tasks requiring deeper linguistic knowledge. Particularly we focus on annotator eye gaze during annotation. Because of recent developments in eye-tracking technology, eye gaze data has been widely used in various research fields, including psycholinguistics and problem solving (Duchowski, 2002). There have been a number of studies on the relations between eye gaze and language comprehension/production (Griffin and Bock, 2000; Richardson et al., 2007). Compared to the studies on language and eye gaze, the role of gaze in general problem solving settings has been less studied (Bednarik and Tukiainen, 2008; Rosengrant, 2010; Tomanek et al., 2010). Since our current interest, text annotation, can be considered a problem solving as well as language comprehension task, we refer to them when defining our prob-

Proceedings of the 7th Linguistic Annotation Workshop & Interoperability with Discourse, pages 214–222,
Sofia, Bulgaria, August 8-9, 2013.

lem setting. Through analysis of annotators' eye-tracking data, we aim at finding useful information which can be employed as features in ML algorithms.

This paper is organised as follows. Section 2 presents the details of the experiment for collecting annotator behavioural data during annotation as well as details on the collected data. Section 3 explains the structure of the annotation process for a single annotation instance. Section 4 provides a detailed analysis of human annotation processes, suggesting usages of those results in NLP. Section 5 reviews the related work and Section 6 concludes and discusses future research directions.

2 Data collection

2.1 Materials and procedure

We conducted an experiment for collecting annotator actions and eye gaze during the annotation of predicate-argument relations in Japanese texts. Given a text in which candidates of predicates and arguments were marked as *segments* (i.e. text spans) in an annotation tool, the annotators were instructed to add links between correct predicate-argument pairs by using the keyboard and mouse. We distinguished three types of links based on the case marker of arguments, i.e. *ga* (nominative), *o* (accusative) and *ni* (dative). For elliptical arguments of a predicate, which are quite common in Japanese texts, their antecedents were linked to the predicate. Since the candidate predicates and arguments were marked based on the automatic output of a parser, some candidates might not have their counterparts.

We employed a multi-purpose annotation tool *Slate* (Kaplan et al., 2012), which enables annotators to establish a link between a predicate segment and its argument segment with simple mouse and keyboard operations. Figure 1 shows a screenshot of the interface provided by *Slate*. Segments for candidate predicates are denoted by light blue rectangles, and segments for candidate arguments are enclosed with red lines. The colour of links corresponds to the type of relations; red, blue and green denote nominative, accusative and dative respectively.

In order to collect every annotator operation, we modified *Slate* so that it could record several important annotation events with their time stamp. The recorded events are summarised in Table 1.

Event label	Description
create_link_start	creating a link starts
create_link_end	creating a link ends
select_link	a link is selected
delete_link	a link is deleted
select_segment	a segment is selected
select_tag	a relation type is selected
annotation_start	annotating a text starts
annotation_end	annotating a text ends

Table 1: Recorded annotation events

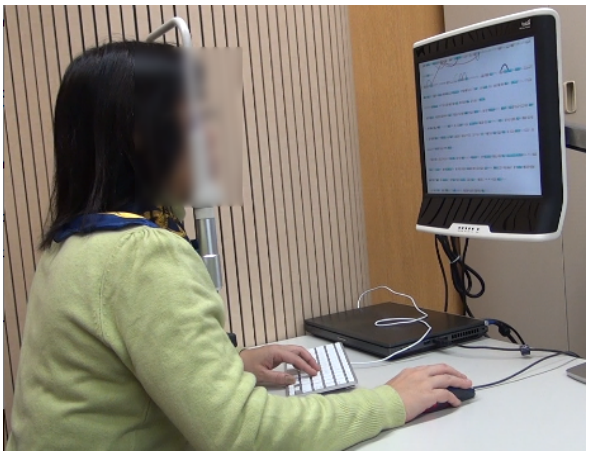

Figure 2: Snapshot of annotation using Tobii T60

Annotator gaze was captured by the Tobii T60 eye tracker at intervals of 1/60 second. The Tobii's display size was $1,280 \times 1,024$ pixels and the distance between the display and the annotator's eye was maintained at about 50 cm. The five-point calibration was run before starting annotation. In order to minimise the head movement, we used a chin rest as shown in Figure 2.

We recruited three annotators who had experiences in annotating predicate-argument relations. Each annotator was assigned 43 texts for annotation, which were the same across all annotators. These 43 texts were selected from a Japanese balanced corpus, BCCWJ (Maekawa et al., 2010). To eliminate unneeded complexities for capturing eye gaze, texts were truncated to about 1,000 characters so that they fit into the text area of the annotation tool and did not require any scrolling. It took about 20–30 minutes for annotating each text. The annotators were allowed to take a break whenever she/he finished annotating a text. Before restarting annotation, the five-point calibration was run every time. The annotators accomplished all assigned texts after several sessions for three or more days in total.

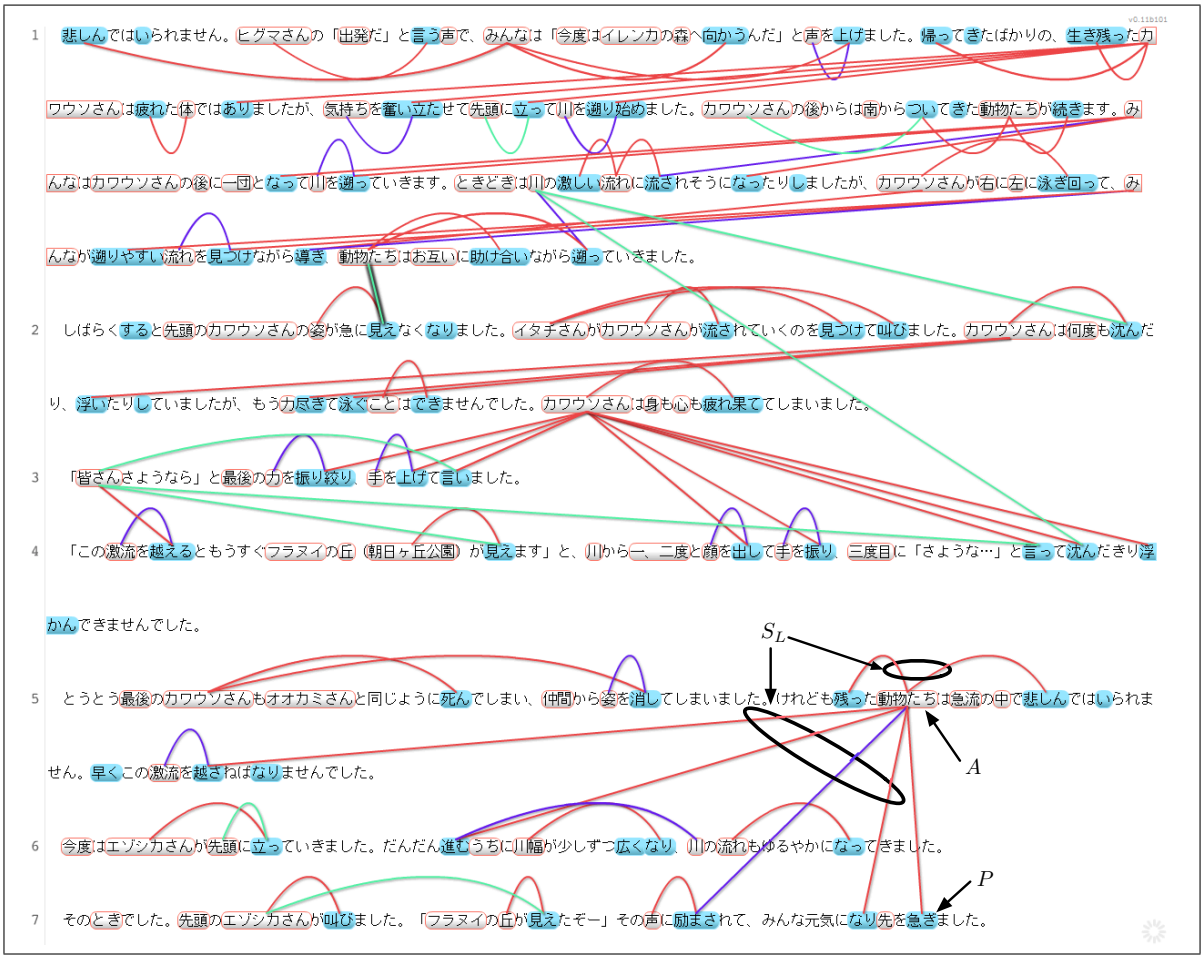

Figure 1: Screenshot of the annotation tool *Slate*

2.2 Results

The number of annotated links between predicates and arguments by three annotators A_0, A_1 and A_2 were 3,353 (A_0), 3,764 (A_1) and 3,462 (A_2) respectively. There were several cases where the annotator added multiple links with the same link type to a predicate, e.g. in case of conjunctive arguments; we exclude these instances for simplicity in the analysis below. The number of the remaining links were 3,054 (A_0), 3,251 (A_1) and 2,996 (A_2) respectively. In addition, because our analyses explained in Section 4 require an annotator's fixation on both a predicate and its argument, the number of these instances were reduced to 1,776 (A_0), 1,430 (A_1) and 1,795 (A_2) respectively. The details of the instances for our analysis are summarised in Table 2. These annotation instances were used for the analysis in the rest of this paper.

case	A_0	A_1	A_2	total
ga (nominative)	1,170	904	1,105	3,179
o (accusative)	383	298	421	1,102
ni (dative)	223	228	269	720
total	1,776	1,430	1,795	5,001

Table 2: Results of annotation by each annotator

3 Anatomy of human annotation

From a qualitative analysis of the annotator's behaviour in the collected data, we found the annotation process for predicate-argument relations could be decomposed into the following three stages.

1. An annotator reads a given text and understands its contents.
2. Having fixed a target predicate, she/he searches for its argument in the set of preceding candidate arguments considering a type of relations with the predicate.
3. Once she/he finds a probable argument in a text, she/he looks around its context in order to confirm the relation. The confirmation is finalised by creating a link between the predicate and its argument.

The strategy of searching for arguments after fixing a predicate would reflect the linguistic knowledge that a predicate subcategorises its arguments. In addition, since Japanese is a head-final language, a predicate basically follows its arguments. Therefore searching for each argument within a sentence can begin at the same position, i.e. the predicate, toward the beginning of the sentence, when the predicate-first search strategy is adopted.

The idea of dividing a cognitive process into different functional stages is common in cognitive science. For instance, Just and Carpenter (1985) divided a problem solving process into three stages: *searching*, *comparison* and *confirmation*. In their task, given a picture of two cubes with a letter on each surface, a participant is instructed to judge whether they can be the same or not. Since one of the cubes is relatively rotated in a certain direction and amount, the participant needs to mentally rotate the cubes for matching. Russo and Leclerc (1994) divided a visual decision making process into three stages: *orientation*, *evaluation* and *verification*. In their experiment, participants were asked to choose one of several daily food products that were visually presented. The boundaries of the above three stages were identified based on the participants' eye gaze and their verbal protocols. Malcolm and Henderson (2009) applied the idea to a visual search process, dividing it into *initiation*, *scanning* and *verification*. Gidlöf et al. (2013) discussed the difference between a decision making process and a visual search process in terms of the process division. Although the above studies deal with the different cognitive processes, it is common that the first stage is for capturing an overview of a problem, the second is for searching for a tentative solution, and the third is for verifying their solution.

Our division of the annotation process conforms with this idea. Particularly, our task is similar to the decision making process as defined by Russo and Leclerc (1994). Unlike these past studies, however, the beginning of an orientation stage[1] is not clear in our case, since we collected the data in a natural annotation setting, i.e. a single annotation session for a text includes creation of multiple links. In other words, the first stage might correspond to multiple second and third stages. In addition, in past research on decision making, a single object is chosen, but our annotation task involves two objects to consider, i.e. a predicate and an argument.

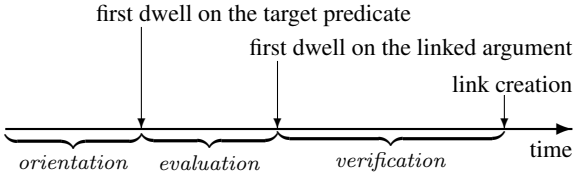

Figure 3: Division of an annotation process

Considering these differences and the proposals of previous studies (Russo and Leclerc, 1994; Gidlöf et al., 2013), we define the three stages as follows. As explained above, we can not identify the beginning of an orientation stage based on any decisive clue. We define the end of an orientation stage as the onset of the first dwell[2] on a predicate being considered. The succeeding evaluation stage starts at the onset of the first dwell on the predicate and ends at the onset of the first dwell on the argument that is eventually linked to the predicate. The third stage, a verification stage, starts at the onset of the first dwell on the linked argument and ends at the creation of the link between the predicate and argument. These definitions and the relations between the stages are illustrated in Figure 3.

The time points indicating the stage boundaries can be identified from the recorded eye gaze and tool operation data. First, gaze fixations were extracted by using the Dispersion-Threshold Identification (I-DT) algorithm (Salvucci and Goldberg, 2000). Based on a rationale that the eye movement velocity slows near fixations, the I-DT algorithm identifies fixations as clusters of consecutive gaze points within a particular dispersion. It has two parameters, the dispersion threshold that defines the maximum distance between gaze points belonging to the same cluster, and the duration threshold that constrains the minimum fixation duration. Considering the experimental configurations, i.e. (i) the display size and its resolution, (ii) the distance between the display and the annotator's eyes, and (iii) the eye-tracker resolution, we set the dispersion threshold to 16 pixels. Following Richardson et al. (2007), we set the duration threshold to 100 msec. Based on fixations, a dwell on a segment was defined as a series of fixations that consecutively stayed on the same segment where

[1] We follow the wording by Russo and Leclerc (1994).

[2] A dwell is a collection of one or several fixations within a certain area of interest, a segment in our case.

two consecutive fixations were not separated by more than 100 msec. We allowed a horizontal error margin of 16 pixels (one-character width) for both sides of a segment when identifying a dwell. Time points of link creation were determined by the "create_link_start" event in Table 1.

Among these three stages, the evaluation stage would be most informative for extracting useful features for ML algorithms, because an annotator identifies a probable argument for a predicate under consideration during this stage. Analysing annotator eye gaze during this stage could reveal useful information for predicate-argument analysis. It is, however, insufficient to regard only fixated arguments as being under the annotator's consideration during the evaluation stage. The annotator captures an overview of the current problem during the previous orientation stage, in which she/he could remember several candidate arguments in her/his short-term memory, then moves on to the evaluation stage. Therefore, all attended arguments are not necessarily observed through gaze dwells. As we explained earlier, we have no means to identify a rigid duration of an orientation stage, thus it is difficult to identify a precise set of candidate arguments under the annotator's consideration in the evaluation stage. For this purpose, we need a different experimental design so that every predicate-argument relation is annotated at a time in the same manner as the above decision making studies conducted. Another possibility is using an annotator's verbal protocols together with her/his eye gaze as done in Russo and Leclerc (1994).

On the other hand, in the verification stage a probable argument has been already determined and its validity confirmed by investigating its competitors. We would expect considered competitors are explicitly fixated during this stage. Since we have a rigid definition of the verification stage duration, it is possible to analyse the annotator's behaviour during this stage based on her/his eye gaze. For this reason, we concentrate on the analysis of the verification stage of annotation henceforth.

4 Analysis of the verification stage

Given the set of annotation instances, i.e. predicate, argument and case triplets, we categorise these instances based on the annotator's behaviour during the verification stage. We focus on two factors for categorising annotation instances: (i) the distance of a predicate and if its argument is either near or far, and (ii) whether annotator gaze dwelled on other arguments than the eventually linked argument before creating the link. We call the former factor *Near/Far* distinction, and the latter *Concentrated/Distracted* distinction.

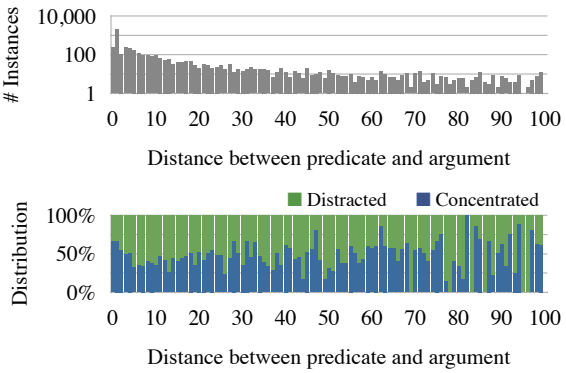

Figure 4: Distance of predicate and argument

To decide the *Near/Far* distinction, we investigated the distribution of distances of predicates and their argument. The result is shown in the upper graph of Figure 4, where the x-axis is the character-based distance and the y-axis shows the number of instances in each distance bin. Figure 4 demonstrates that the instances concentrate at the bin of distance 1. This reflects the frequently occurring instances where a one-character case maker follows an argument, and immediately precedes its predicate. The lower graph in Figure 4 shows the ratio of *Distracted* instances to *Concentrated* at each bin. The distribution indicates that there is no remarkable relation between the distance and *Concentrated/Distracted* distinction. The correlation coefficient between the distance and the number of *Concentrated* instances is -0.26. We can conclude that the distance of a predicate and its argument does not impact the *Concentrated/Distracted* distinction. Considering the above tendency, we set the distance threshold to 22, the average distance of all annotation instances; instances with a distance of less than 22 are considered *Near*.

These two factors make four combinations in total, i.e. *Near-Concentrated* (NC), *Near-Distracted* (ND), *Far-Concentrated* (FC) and *Far-Distracted* (FD). We analysed 5,001 instances shown in Table 2 to find three kinds of tendencies, which are described in the following sections.

case	*Near*	*Far*	total
ga (nominative)	2,201 (0.44)	978 (0.90)	3,179 (0.64)
o (accusative)	1,042 (0.34)	60 (0.05)	1,102 (0.22)
ni (dative)	662 (0.22)	58 (0.05)	720 (0.14)

Table 3: Distribution of cases over *Near/Far*

	NC	ND	FC	FD
ga	0.40	0.47	0.92	0.90
o, ni	0.60	0.53	0.08	0.10

Table 4: Distribution of arguments across four categories

4.1 Predicate-argument distance and argument case

We hypothesise that an annotator changes her/his behaviour with regard to the case of the argument. The argument case in Japanese is marked by a case marker which roughly corresponds to the argument's semantic role, such as Agent and Theme. We therefore analysed the relationship between the *Near*/*Far* distinction and argument case. The results are shown in Table 3. The table shows the distribution of argument cases, illustrating that *Near* instances are dispersed over three cases, while *Far* instances are concentrated in the *ga* (nominative) case. In other words, *ga*-arguments tend to appear far from their predicate. This tendency reflects the characteristic of Japanese where a nominative argument tends to be placed in the beginning of a sentence; furthermore, *ga*-arguments are often omitted to make ellipses. In our annotation guideline, a predicate with an elliptical argument should be linked to the referent of the ellipsis, which would be realised at a further distant position in the preceding context. In contrast, *o* (accusative) and *ni* (dative) arguments less frequently appeared as *Far* instances because they are rarely omitted due to their tighter relation with arguments. This observation suggests that each case requires an individual specific treatment in the model of predicate argument analysis; the model searches for *o* and *ni* arguments close to its predicate, while it considers all preceding candidates for a *ga* argument.

Table 4 shows the break down of the *Near/Far* columns with regards to the *Concentrated*/*Distracted* distinction, demonstrating that the *Concentrated*/*Distracted* distinction does not impact the distribution of the argument types.

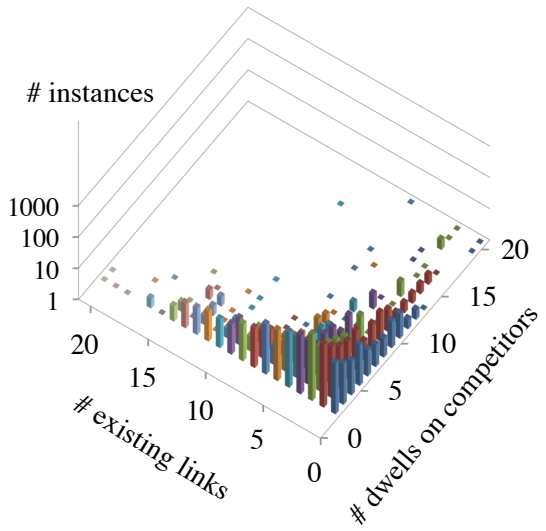

Figure 5: Relationship between the number of dwells on competitors and already-existing links

4.2 Effect of already-existing links

In the *Concentrated* instances, an annotator can verify if an argument is correct without inspecting its competitors. As illustrated in Figure 1, already annotated arguments are marked by explicit links to their predicate. These links make the arguments visually as well as cognitively salient in an annotator's short-term memory because they have been frequently annotated in the preceding annotation process. Thus, we expected that both types of saliency help to confirm the predicate-argument relation under consideration. For instance, when searching for an argument of predicate P in Figure 1, argument A that already has six links (S_L) is more salient than other competitors.

To verify this hypothesis, we examined the relation of the number of already-existing links and the number of dwells on competitors, which is shown in Figure 5. In this analysis, we used only *Far* instances because the *Near* arguments tended to have less already-existing links as they were under current interest. Figure 5 shows a three-dimensional declining slope that peaks around the intersection for instances with the fewest number of links and dwells on competitors. It reveals a mostly symmetrical relation between existing links and dwells on competitors for instances with a lower number of existing links, but that this symmetry brakes for instances with a higher number of existing links, visible by the conspicuous hole

toward the left of the figure. This suggests that visual and cognitive saliency reduces annotators' cognitive load, and thus contributes to efficiently confirming the correct argument.

This result implies that the number of already-existing links of a candidate argument would reflect its saliency, thus more linked candidates should be preferred in the analysis of predicate-argument relations. Although we analysed the verification stage, the same effect could be expected in the evaluation stage as well. Introducing such information into ML algorithms may contribute to improving system performance.

4.3 Specificity of arguments and dispersal of eye gaze

Existing Japanese corpora annotated with predicate-argument relations (Iida et al., 2007; Kawahara et al., 2002) have had syntactic heads (nouns) of their projected NPs related their predicates. Since Japanese is a head-final language, a head noun is always placed in the last position of an NP. This scheme has the advantage that predicate-argument relations can be annotated without identifying the starting boundary of the argument NP under consideration. The scheme is also reflected in the structure of automatically constructed Japanese case frames, e.g. Sasano et al. (2009), which consist of triplets in the form of ⟨*Noun, Case, Verb*⟩. *Noun* is a head noun extracted from its projected NP in the original text. We followed this scheme in our annotation experiments.

However, a head noun of an argument does not always have enough information. A nominaliser which often appears in the head position in an NP does not have any semantic meaning by itself. For instance, in the NP "*benkyô suru koto* (to study/studying)", the head noun "*koto*" has no specific semantic meaning, corresponding to an English morpheme "to" or "-ing". In such cases, inspecting a whole NP including its modifiers is necessary to verify the validity of the NP for an argument in question. We looked at our data to see if annotators actually behaved like this.

For analysis, the annotation instances were distinguished if an argument had any modifier or not (column "w/o mod" and "w/ mod" in Table 5). The "w/ mod" instances are further divided into two classes: "within NP" and "out of NP", the former if all dwells remain "within" the region of the argument NP or the later if they go "out of" the region. Note that our annotation scheme creates a link between a predicate and the head of its argument as described earlier. Thus, a *Distracted* instance does not always mean an "out of NP" instance, since a distracted dwell might still remains on a segment within the NP region despite not being its head. Table 5 shows the distribution of the instances over this categorisation.

	w/o mod	w/ mod		total
		within NP	out of NP	
Concentrated	1,562	1190	–	2,752
Distracted	1,168	242	839	2,249

Table 5: Relation of argument modifiers and gaze dispersal

We found that the number of instances is almost the same between *Concentrated* and *Distracted*, i.e. (2752 : 2249 = 0.55 : 0.45). In this respect, both *Concentrated* and *Distracted* instances can be treated in the same way in the analysis of predicate-argument relations. A closer look at the break down of the "w/ mod" category, however, reveals that almost 22% of the *Distracted* arguments with any modifier attracted gaze dwells within the NP region. This fact suggests that we need to treat candidate arguments differently depending on if they have modifiers or not. In addition to argument head information, we could introduce information of modifiers into ML algorithms as features that characterise a candidate argument more precisely.

5 Related work

Recent developments in the eye-tracking technology enables various research fields to employ eye-gaze data (Duchowski, 2002).

Bednarik and Tukiainen (2008) analysed eye-tracking data collected while programmers debug a program. They defined areas of interest (AOI) based on the sections of the integrated development environment (IDE): the source code area, the visualised class relation area and the program output area. They compared the gaze transitions among these AOIs between expert and novice programmers to find different transition patterns between them. Since the granularity of their AOIs is coarse, it could be used for evaluating a programmer's expertise, but hardly explains why the expert transition pattern realises a good programming skill. In order to find useful information for language processing, we employed smaller AOIs

at the character level.

Rosengrant (2010) proposed an analysis method named *gaze scribing* where eye-tracking data is combined with a subject's thought process derived by the think-aloud protocol (TAP) (Ericsson and Simon, 1984). As a case study, he analysed a process of solving electrical circuit problems on the computer display to find differences of problem solving strategy between novice and expert subjects. The AOIs are defined both at a macro level, i.e. the circuit, the work space for calculation, and a micro level, i.e. electrical components of the circuit. Rosengrant underlined the importance of applying gaze scribing to the solving process of other problems. Although information obtained from TAP is useful, it increases her/his cognitive load, and thus might interfere with her/his achieving the original goal.

Tomanek et al. (2010) utilised eye-tracking data to evaluate the degree of difficulty in annotating named entities. They are motivated by selecting appropriate training instances for active learning techniques. They conducted experiments in various settings by controlling characteristics of target named entities. Compared to their named entity annotation task, our annotation task, annotating predicate-argument relations, is more complex. In addition, our experimental setting is more natural, meaning that all possible relations in a text were annotated in a single session, while each session targeted a single named entity (NE) in a limited context in the setting of Tomanek et al. (2010). Finally, our fixation target is more precise, i.e. words, rather than a coarse area around the target NE.

We have also discussed evaluating annotation difficulty for predicate-argument relations by using the same data introduced in this paper (Tokunaga et al., 2013). Through manual analysis of the collected data, we suggested that an annotation time necessary for annotating a single predicate-argument relation was correlated with the agreement ratio among multiple human annotators.

6 Conclusion

This paper presented an analysis of an annotator's behaviour during her/his annotation process for eliciting useful information for NLP tasks. We first conducted an experiment for collecting three annotators' actions and eye gaze during their annotation of predicate-argument relations in Japanese texts. The collected data were analysed from three aspects: (i) the relationship of predicate-argument distances and argument's cases, (ii) the effect of already-existing links and (iii) specificity of arguments and dispersal of eye gaze. The analysis on these aspects suggested that obtained insight into human annotation behaviour could be useful for exploring effective linguistic features in ML-based approaches.

As future work, we need to further investigate the data from other aspects. There are advantages to manual analysis, such as done in this paper. Mining techniques for finding unknown but useful information may also be advantageous. Therefore, we are planning to employ mining techniques for finding useful gaze patterns for various NLP tasks.

In this paper, we suggested useful information that could be incorporated into ML algorithms as features. It is necessary to implement these features in a specific ML algorithm and evaluate their effectiveness empirically.

Our analysis was limited to the verification stage of annotation, in which a probable argument of a predicate was confirmed by comparing it with other competitors. The preceding evaluation stage should be also analysed, since it is the stage where annotators search for a correct argument of a predicate in question, thus probably includes useful information for computational models in identifying predicate-argument relations. For the analysis of the evaluation stage, a different design of experiments would be necessary, as already mentioned, employing single annotation at a time scheme as Tomanek et al. (2010) did, or using an annotator's verbal protocol together as Russo and Leclerc (1994), and Rosengrant (2010) did.

Last but not least, data collection and analysis in different annotation tasks are indispensable. It is our ultimate goal to establish a methodology for collecting an analysing annotators' behavioural data during annotation in order to elicit effective features for ML-based NLP.

References

Roman Bednarik and Markku Tukiainen. 2008. Temporal eye-tracking data: Evolution of debugging strategies with multiple representations. In *Proceedings of the 2008 symposium on Eye tracking research & applications (ETRA '08)*, pages 99–102.

Andrew T. Duchowski. 2002. A breadth-first survey of eye-tracking applications. *Behavior Research Meth-*

ods, Instruments, and Computers, 34(4):455–470.

David duVerle and Helmut Prendinger. 2009. A novel discourse parser based on support vector machine classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 665–673.

K. Anders Ericsson and Herbert A. Simon. 1984. *Protocol Analysis – Verbal Reports as Data –*. The MIT Press.

Kerstin Gidlöf, Annika Wallin, Richard Dewhurst, and Kenneth Holmqvist. 2013. Using eye tracking to trace a cognitive process: Gaze behaviour during decision making in a natural environment. *Journal of Eye Movement Research*, 6(1):1–14.

Zenzi M. Griffin and Kathryn Bock. 2000. What the eyes say about speaking. *Psychological Science*, 11(4):274–279.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Ryu Iida, Mamoru Komachi, Kentaro Inui, and Yuji Matsumoto. 2007. Annotating a Japanese text corpus with predicate-argument and coreference relations. In *Proceeding of the ACL Workshop 'Linguistic Annotation Workshop'*, pages 132–139.

Marcel Adam Just and Patricia A. Carpenter. 1985. Cognitive coordinate systems: Accounts of mental rotation and individual differences in spatial ability. *Psychological Review*, 92(2):137–172.

Dain Kaplan, Ryu Iida, Kikuko Nishina, and Takenobu Tokunaga. 2012. Slate – a tool for creating and maintaining annotated corpora. *Journal for Language Technology and Computational Linguistics*, 26(2):89–101.

Daisuke Kawahara, Sadao Kurohashi, and Kôiti Hasida. 2002. Construction of a Japanese relevance-tagged corpus (in Japanese). In *Proceedings of the 8th Annual Meeting of the Association for Natural Language Processing*, pages 495–498.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2010. A PDTB-styled end-to-end discourse parser. Technical Report TRB8/10, School of Computing, National University of Singapore.

Kikuo Maekawa, Makoto Yamazaki, Takehiko Maruyama, Masaya Yamaguchi, Hideki Ogura, Wakako Kashino, Toshinobu Ogiso, Hanae Koiso, and Yasuharu Den. 2010. Design, compilation, and preliminary analyses of balanced corpus of contemporary written Japanese. In *Proceedings of the Eigth International Conference on Language Resources and Evaluation (LREC 2010)*, pages 1483–1486.

George L. Malcolm and John M. Henderson. 2009. The effects of target template specificity on visual search in real-world scenes: Evidence from eye movements. *Journal of Vision*, 9(11):8:1–13.

George A. Miller. 1995. WordNet: A lexical database for English. *Communications of the ACM*, 38:39–41.

Vincent Ng. 2010. Supervised noun phrase coreference research: The first fifteen years. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010)*, pages 1396–1411.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL – Shared Task*, pages 1–40.

Daniel C. Richardson, Rick Dale, and Michael J. Spivey. 2007. Eye movements in language and cognition: A brief introduction. In Monica Gonzalez-Marquez, Irene Mittelberg, Seana Coulson, and Michael J. Spivey, editors, *Methods in Cognitive Linguistics*, pages 323–344. John Benjamins.

David Rosengrant. 2010. Gaze scribing in physics problem solving. In *Proceedings of the 2010 symposium on Eye tracking research & applications (ETRA '10)*, pages 45–48.

J. Edward Russo and France Leclerc. 1994. An eye-fixation analysis of choice processes for consumer nondurables. *Journal of Consumer Research*, 21(2):274–290.

Dario D. Salvucci and Joseph H. Goldberg. 2000. Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications (ETRA '00)*, pages 71–78.

Ryohei Sasano, Daisuke Kawahara, and Sadao Kurohashi. 2009. The effect of corpus size on case frame acquisition for discourse analysis. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2009)*, pages 521–529.

Takenobu Tokunaga, Ryu Iida, and Koh Mitsuda. 2013. Annotation for annotation - toward eliciting implicit linguistic knowledge through annotation -. In *Proceedings of the 9th Joint ISO - ACL SIGSEM Workshop on Interoperable Semantic Annotation (ISA-9)*, pages 79–83.

Katrin Tomanek, Udo Hahn, Steffen Lohmann, and Jürgen Ziegler. 2010. A cognitive cost model of annotations based on eye-tracking data. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010)*, pages 1158–1167.