Pair Annotation: Adaption of Pair Programming to Corpus Annotation

Işın Demirşahin
Cognitive Science Program
Middle East Technical University
Ankara, Turkey
disin@metu.edu.tr

İhsan Yalçınkaya
Cognitive Science Program
Middle East Technical University
Ankara, Turkey
yalcinkaya.ihsan@gmail.com

Deniz Zeyrek
Cognitive Science Program
Middle East Technical University
Ankara, Turkey
dezeyrek@metu.edu.tr

Abstract

This paper will introduce a procedure that we call pair annotation after pair programming. We describe initial annotation procedure of the TDB, followed by the inception of the pair annotation idea and how it came to be used in the Turkish Discourse Bank. We discuss the observed benefits and issues encountered during the process, and conclude by discussing the major benefit of pair annotation, namely higher inter-annotator agreement values.

1 Introduction

The Turkish Discourse Bank (TDB) is a 500,000-word subcorpus of METU Turkish Corpus (Say et al., 2002), which is annotated for discourse connectives in the style of Penn Discourse Tree Bank (PDTB) (Prasad et al., 2008). As in the PDTB; discourse connectives are annotated along with the two text spans they link. The text spans can be single or multiple verb phrases, clauses, or sentences that can be interpreted as abstract objects (Asher, 1993). The text span that syntactically hosts the connective is labeled the second argument (Arg2), while the other text span is labeled the first argument (Arg1). The TDB annotations were carried out using the Discourse Annotation Tool for Turkish (DATT) (Aktaş, et al., 2010). In the first release of TDB, a total of 8482 relations are annotat-annotated for 147 connectives.[1]

In this paper, we first describe the initial annotation procedure. Then, we introduce how the pair annotation (PA) procedure emerged. We continue with agreement statistics on some connectives annotated via the PA procedure, and finally we discuss the advantages and disadvantages of PA.

2 Initial Annotation Procedure in the TDB

2.1 Independent Annotations

The initial step in the TDB project was to determine which instances of the connectives would be annotated as discourse connectives. First, the connective tokens were revealed. Using each token as a search unit, three annotators went through the whole corpus and annotated all discourse connective instances of the token, together with the texts spans they link. Each annotator worked individually and independently, and did not have access to the annotations of other annotators.

Some search units corresponded to several discourse connectives. For example, the search for the unit *halde* 'although, despite' results in four discourse connectives. It appears as a complex subordinator that expects a nominalizing suffix -DIK and a case marker on its second argument as in (1). In the examples, Arg1 is shown in italics, and Arg2 is set in bold. The connective is underlined and the modifier is in square brackets when present.

(1) **Doğu Beyazıt'da geceledięimiz** <u>halde</u> *bir dünya şaheseri olan İshak Paşa medresesini göremeden Ankara'ya döndük.*

[1] The current TDB release can be requested online from the project website at http://medid.ii.metu.edu.tr

Proceedings of the 6th Linguistic Annotation Workshop, pages 31–39,
Jeju, Republic of Korea, 12-13 July 2012. ©2012 Association for Computational Linguistics

Although **we spent the night in Doğu Beyazıt**, *we returned to Ankara without seeing the İshak Paşa Medresseh, which is a masterpiece*.

It also appears with anaphoric elements: *o halde* ‘then, in that case’ as in (2), and *şu halde* ‘in the current situation, in this specific case’ as in (3).

(2) *Beyin delgi ameliyatı, hangi gerekçeyle yapılırsa yapılsın, insanoğlunun gerçekleştirdiği ilk cerrahi müdahaledir*. O halde, **nöroşirürjiyi Neolitik Çağ'a, hatta Mezolitik Çağ'a kadar götürebiliriz**.
Trepanation operations, regardless of the justifications for which they have been carried out, are the first surgical operation ever attempted by mankind. Then, **we can trace neurosurgery back to Neolithic Era, even to Mesolithic Era**.

(3) *Bu seçim, eskisinin devamı niteliğinde olsaydı, 60 günlük bir süreye ihtiyaç duyulmaması ve en kısa zamanda seçime gidil-mesi gerekirdi*. Şu halde **60 günlük süre yeni bir seçimin yapılması için gerekli prosedürün uygulanması ve hazırlıkların tamamlanmasını sağlamak için öngörül-müş bir süredir.**
If the nature of this election was the continuation of the old one, a period of 60 days wouldn’t have been necessary and and the elections would have to be held immediately. In the current situation, **the 60-day period is the anticipated period for the application of the necessary procedure and the completion of the preperations.**

Finally, it appears with the adjective *aksi* ‘opposite’ to form *aksi halde* ‘otherwise’ as in (4).

(4) Feyzi Bey *böyle bir durumda mebusluktan istifa edeceğini*, aksi halde [de] **Falih Rıfkı Bey'in istifa etmesi gerektiğini** belirtmiş.
Mr. Fevzi stated that *in such a situation he would resign from parliament membership*, otherwise **Mr. Falih Rıfkı would have to resign.**

All such occurrences were annotated when searching with the unit *halde*, but are counted as different discourse connectives. There is no label for instances of search units that are not discourse connectives, so all other occurrences were left unannotated. For example, the adverbial clause forming *bir halde* ‘in such a manner’ in (5), which takes the clause *ne yapacağımı bilmez-* ‘doesn’t know what I will do’ and builds and the adverbial clause *ne yapacağımı bilmez bir halde* ‘not knowing what to do’, was not annotated at all.

(5) O gün akşama kadar ne yapacağımı bilmez bir halde dolaştım evin içinde.
I walked around the house till evening that day, not knowing what to do.

2.2 Agreement Procedure

Upon the completion of independent annotations, disagreements were determined and brought to agreement meetings. The agreement meetings were open to the whole research group, which included four researchers in addition to the three annotators. All researchers, annotator and non-annotator, were native speakers of Turkish. In any given agreement meeting at least one non-annotator researcher and at least two annotators were present.

The preferred method for agreement was discussion among the annotators and researchers. The final annotation was not necessarily selected from the independent annotations. Sometimes a partial or complete combination of different annotations was agreed upon, and in few cases, a novel annotation emerged as the agreed annotation.

In cases where the discussion proved to be inconclusive, a non-annotator adjudicator decided how the agreed annotation should be. The adjudicator was constant throughout the project, and had the deepest and most thorough understanding of the annotation guidelines among the research group. When deciding on the agreed annotation, the adjudicator took the preceding consideration into account, as well as the native speaker intuitions of the annotators and the researchers. The adjudicator sometimes consulted the majority vote of the annotations or the research group, but only as long as the majority vote was completely in accord with the annotation guidelines.

The agreement meetings sometimes resulted in additions and/or changes to the annotation guidelines. In such cases, all annotations were checked to preserve consistency across annotations, and the final version was produced, which may be referred to as the gold standard.[2]

2.3 Common Divergences among Independent Annotations

There are five common types of divergence in the annotations.

(a) The first case is a physical error in selecting a connective or argument span. The annotation guidelines state that all the punctuation marks and spaces around argument spans and discourse connectives should be left out, with one exception: when one of a pair of quotation marks or dashes is in the argument span, the matching one is included in the span, too. Sometimes space characters or punctuation marks that should be left out are included in the selection or a quotation mark or dash is excluded although its pair is in the annotated span; or one or more letters of a word is not selected. These errors arise because the DATT allows continuous selection of text and do not snap to word boundaries automatically. The tool is designed in this way so as to allow the annotation of simple subordinators which are single suffixes such as *–dan* in (6).

(6) **Başka kimse olmadığın**dan *iki kadının da yüzü açıktı*.
‘Since **there was no one else**, *the faces of both women were unveiled*.’

In some cases, type (a) divergences occur in a larger scale. The annotation guidelines exclude some text spans such as salutations, commentaries and parenthetical form arguments when they are not vital to the understanding of the discourse connection between the two arguments. Sometimes annotators may overlook this rule and include intrusions of several words that should not be in the argument. Since these cases are explicitly ruled out by the annotation guidelines, these divergences are taken to be errors, rather than genuine cases of disagreement.

(b) The second type of divergence arises when the annotators more or less agree on what the arguments are, but there is a syntactic or semantic ambiguity in the text that prevents them from agreeing on the argument span. For example one annotator may include a temporal adverb in an argument whereas the other annotates the same adverb as “shared”, i.e. applying to both arguments. Similarly, some adverbs like *salt* ‘only, just’ may be understood by one annotator to take an argument as its scope and thus should be included in that argument (7a), whereas the same adverb is considered by another annotator to take the connective as its scope, and as a result it might be annotated as a modifier (7b).

(7a) **Salt gülmek** için *gelmişlerdi*.
They came to **just laugh**.

(7b) [Salt] **gülmek** için *gelmişlerdi*.
They came [only] to **laugh**.

(c) A third type of divergence occurs when the annotators annotate relations differently, because they get different meanings from that part of the text. In these cases, the span annotated by one of the annotators might include, overlap with, or completely differ from the spans of the other annotators, as in (8a) and (8b). (8a) shows that the annotator interpreted the temporal sequence as between the speech and the moving of the funeral, whereas (8b) shows that another annotator believed that the relation was between the ceremony and the moving of the funeral.

(8a) Usumi için ilk tören, Türkiye Gazeteciler Cemiyeti (TGC) önünde düzenlendi. *TGC Başkanı Orhan Erinç, konuşmasında Usumi'nin yokluğunu hissedeceklerini vurguladı.* **Usumi'nin cenazesi** [daha] sonra **Sultanahmet Camii'ne götürüldü**.
The first ceremony for Usumi was arranged in front of the Association of the Journalists of Turkey (TGC). *Orhan Erinç, the chairman the TGC, emphasized that Usumi would be missed*. Then, **the Usumi’s funeral was moved to the Sultan Ahmed Mosque**.

[2] This so called gold standard is suitable for linguistic explorations and descriptive studies as it tries to be as exhaustive as possible, i.e., all located instances of discourse connectives were annotated, including even the hardest of the cases. The inclusion of the hard cases in the data may increase the noise, requiring caution when using the data for benchmarking (Reidsma and Carletta, 2008). A benchmarking version of TDB that excludes the hard cases and includes the noise levels following Klebanov and Beigman (2008) is in preparation.

(8b) *Usumi için ilk tören, Türkiye Gazeteciler Cemiyeti (TGC) önünde düzenlendi*. TGC Başkanı Orhan Erinç, konuşmasında Usumi'nin yokluğunu hissedeceklerini vurguladı. **Usumi'nin cenazesi** [daha] sonra **Sultanahmet Camii'ne götürüldü**.
The first ceremony for Usumi was arranged in front of the Association of the Journalists of Turkey (TGC). Orhan Erinç, the chairman the TGC, emphasized that Usumi would be missed. Then, **the Usumi's funeral was moved to the Sultan Ahmed Mosque**.

Divergences of type (b) and (c) are cases of genuine disagreement, pointing to hard cases; whereas type (a) is a simple case of human error and may arise even in the easiest cases.

(d) Another type of divergence is the case when one or more annotators did not annotate an instance of the search unit, whereas the others have annotated it. This might be because one annotator believed this specific instance of the search unit to be a non-discourse connective, or it might simply be overlooked. The former cases are genuine disagreements, whereas the latter cases are errors in annotation.

(e) The last type of divergence emerges due to cases underdetermined in annotation guidelines. An example of this type of divergence resulted from the case of shared copula during the annotation of *ve* 'and'.

(9) Kızın saçları *siyah* ve **kıvırcıktı**.
The girl's hair was *black* and **curly**.

Because in present tense the copula is often dropped and *Kızın saçları siyah* 'The girl's hair is black' is interpreted as an abstract object, (9) was interpreted by some annotators and researchers as coordination of two abstract objects; whereas others interpreted it as simple adjective coordination, where *ve* 'and' links the two adjectives *siyah* 'black' and *kıvırcık* 'curly'.

During the annotation phase, the guidelines were not clear concerning instances like (9) and were only finalized after further consideration and more exposure to data. Obviously, such underdetermination by annotation guidelines can and does result in major disagreements. However, since this type of disagreement should be settled in the guidelines and cannot be improved by the annotators, type (e) divergences will not be considered further in this paper.

The divergences resulting form human errors were the easiest to resolve during the agreement meetings. Of the genuine disagreements, types (b) and (c) were the harder to resolve because they resulted from ambiguities, and in some cases various annotations seemed plausible.

It was during this discussion of hard cases when the annotators came up with the need to incorporate some sort of discussion into the annotation procedure. When the inter-annotator reliability among three annotators stabilized, it was proposed to use a pair of annotators to carry out the task together while the third annotator continued her task independently in an attempt to accelerate the annotations. This team approach quickly led to the procedure we call pair annotation after the pair programming procedure in software engineering.

3 Pair Programming

Pair programming (PP), also referred to as collaborative programming, is the process where two programmers work together at the same piece of algorithm or code (Williams, et al, 2000; Williams and Kessler, 2000). PP can be taken as a method for software development by itself (Williams and Kessler, 2003), or it can be integrated into other development schemes as in the case of extreme programming (XP) (Beck, 2000).

In pair programming, one of the programmers, the driver, is responsible for physically producing the code or the algorithm. The driver is the one that uses the keyboard to actually write the code. The other programmer, the navigator, continuously monitors the driver and actively takes part in the creation of the code by watching for errors, thinking of alternative strategies or better ways to implement the algorithm and looking up resources that might be needed during coding.

The division of labor and the fact that the driver is the one actually producing the code, however, do not imply that the driver takes a leading role, or has greater part in ownership or responsibility. The ownership of the piece of code developed in PP, and the responsibility for the errors it may contain, belong to both programmers equally. Therefore, the navigator needs to be actively involved at all times, for they will be held equally responsible if anything goes wrong. The role of the driver is switched periodically, so in the overall process, both programmers have equal roles as well as equal credit and equal responsibility.

3.1 Advantages of Pair Programming

Programmers observe that when they work in pairs, they produce higher quality software in less time than it would take to produce the software by means of individual programming. They also report that they have higher motivation while programming, because they feel responsibility towards their partner. They put less time in irrelevant or personal tasks and concentrate more on the job at hand because they feel otherwise they would be wasting their partner's time. In addition, working together with a partner and creating a jointly owned product brings the programmers close, leading to a case called pair jelling (Williams, et al., 2000) in which "the sum is greater than its parts" (DeMarco and Lister, 1977, as cited in Williams, et al, 2000), which in turn facilitates the pair performance to exceed the performance of the individual programmers, or even their individual performances combined.

One of the major costs in the budget of a project, and an often overlooked one, is the time spent for communication between the teams or programmers who take part in the development of software. The cost of the project is usually calculated on the basis of programmer hours and these hours usually indicate only the actual coding hours, but Brooks (1975) states that the time spent for communication should also be included in the overall cost of the project. Williams and Kessler (2000) report that PP decreases this communication time thanks to the already established communication channels and protocols within the programming pair.

3.2 Disadvantages of Pair Programming

The fact that PP takes a shorter period of time to produce a piece of software does not mean that it takes up less resource. The most prominent disadvantage of PP for those who encounter the idea the first time is that it is a waste of time to put two programmers to a job that could have been carried out by only one. Even if the software is produced quicker than when it was produced by an individual programmer, the overall programmer hours is expected to be so high that the procedure is not likely to be cost efficient. Research shows that this is not necessarily the case. Although it might take more time to complete a task compared to individual programmers when the programmers are newly introduced to the PP, as they become more experienced, the overall programming hours spent on the task come close to time spent when the programming is done individually.

In one case, when the programmers are first introduced to PP, the pairs completed a task faster and more accurately then individual programmers, but the overall programming hours was 60% higher that individual programmers. However, as the programmers adapted to the procedure, the increase was reduced to 15% (Williams, et al., 2000). Considering the fact that a less accurate code will need much debugging, this 15% increase in programming time seems to be acceptable.

4 Pair Annotation

To keep the annotations as unbiased as possible while accelerating the annotation process, the TDB group decided to keep one of the individual annotators independent. Two other annotators teamed up and annotated as a pair, which would be treated as a single annotator in the agreement process.

At the time of the introduction of the pair annotation (PA) procedure to the project, two of the annotators had some degree of familiarity with the idea of pair programming; but it did not immediately occur to them to relate software programming and corpus annotation processes. As pair annotation advanced, the most basic principles of PP emerged on their own accord. It was more practical to let one of the annotators handle the input for the whole session, so the roles of the driver and navigator arose. The corrective and the supportive role of the navigator also emerged because of the self-imposed responsibility of the person who was not actually handling the keyboard-mouse. She neither wanted to leave the entire job to the other person, nor to be left out of the annotation process. For similar reasons, switching of the driver/navigator roles followed. As the PA routine became more and more established, the similarities between the PA and PP routines became more prominent.

The agreement process for PA is similar to independent annotations; but the pair is treated as a single entity, especially where majority vote is concerned. The annotators in the pair are free to voice their opinions; however, care is taken to prevent the pair from biasing the gold standard.

4.1 Observed Benefits of Pair Annotation

During the PA experience, the annotators observed that the frequency of errors, especially that of type (a) decreased; because even if the driver made as many mistakes as an individual annotator, the navigator almost always warned her. The mistake was immediately corrected, and therefore they would not appear as disagreements in later phases.

When done in pairs, annotation of the hard cases of type (b) and (c) was faster, too. Sometimes the pair had to carry out lengthy discussions until they agreed on an annotation. Although this seems like it might prolong the annotation time, it did not.

In the cases when a relation is hard to annotate due to ambiguities, all individual annotators would spend a long time on the same relation to understand the larger context to resolve the ambiguity. Sometimes they would have to recall, or search for, a piece of background knowledge that is necessary to process the text. In fact, it usually takes an individual longer than a pair to complete such difficult annotations because the pair can search twice as fast, as well as sharing their knowledge about the context, sometimes eliminating the need to spend any time at all. As a result, a pair annotates a set of relations faster than an individual does.

As mentioned in the PP literature, yet another benefit is higher motivation during annotation. Annotating the same connective in the corpus can sometimes become a tedious job, but having a partner to discuss cases, or even just share complaints or jokes lightens up the process considerably. A repetitive and tedious job becomes interactive and even enjoyable. Moreover, similar to PP, pair annotations are done more efficiently because the partners spend less time on unrelated or personal activities during the designated PA times due to the fact that they do not want to waste each other's time.

In addition to decreasing the time spent during annotation, PA decreases the overall time spent on the agreement procedures just as PP decreases the time spent on communication between programmers. In those cases when the pair have already discussed a particular annotation, they summarize the results of this discussion in a notes field provided in the annotation tool. These discussion summaries present their justification for their annotation to the research group during the agreement meeting. Although the notes field contributes to agreement of individual annotations in a similar manner, the notes of a pair include the already compared and evaluated views of two annotators and a proposed resolution, which results in agreement in a shorter time.

4.2 Issues in Pair Annotation

Questions arise against PA similar to those that arise against PP. Is it not a waste of time to ask three annotators to work if all we are going to have are two sets of annotations? If we can put three annotators to the job, is it not preferable to have three sets of annotations instead of two? From one point of view, the more sets of independent annotations, the better. However, it is common practice for corpus annotation projects to decrease the number of annotators once disagreement stabilizes, as in the example of the PDTB (Miltsakaki, et al, 2004) and it is this practice that we adopted in the TDB.

Another concern that arises for both PP and PA is what if one partner -the usual candidate is the navigator- does not participate in the process actively? Or what if one partner constantly dominates the process and ignores the opinions of the other? The TDB has not encountered this specific problem mainly because the annotators have been involved in the process from the beginning of the project, and have taken active roles in building the annotation principles. In other projects where certain annotators have to contribute for a limited amount of time only, this may become an important caveat. To circumvent the potential problem, the pairs might be asked for feedback periodically to make sure that the PA procedure is working as intended.

Finally, there are annotation specific questions concerning PA. There is always the threat that a pair's annotation could be biased, because the pair interacts constantly. As a result of their discussions or the persuasive powers of one of the partners, the resulting annotations may diverge from the initial native speaker intuitions of the annotators; or while trying to combine two different annotations, the result may end up being counterintuitive. In the TDB, we did not come across this problem thanks to the productive utilization of the notes field.

As explained above, the annotators use the notes field to summarize their discussions of the hard cases. By doing so, they include the first intuitions of both annotators and the reasoning process of their resulting annotation. In some cases they use the field to declare that a joint annotation could not be reached. These comments have been very useful during the agreement meetings for the pair annotation and also contributed to the improvement of annotation schema and annotation guidelines.

Pair annotation is not the solution to all problems in annotation, nor does it offer the perfect annotation procedure. That is why what we propose here is not replacing the entire annotation progress with PA, but having an independent individual annotator in addition to the pair. The procedure we are describing is closer to having two independent annotators, where one of the annotators is like a composite being consisting of two individuals thinking independently, but producing a single set of annotations collaboratively. Similar to the joint ownership of PP, neither partner claims the annotation as her own, but the annotation is treated as it belongs to a single annotator, i.e. the pair. It is treated as a single set of annotations both during the agreement meetings and in calculating the agreement statistics.

4.3 Effect of Pair Annotation on the Agreement Statistics

For four high frequency connectives, *ama* ‘but’, *sonra* ‘after’, *ve* ‘and’ and *ya da* ‘or’, the first 1/3 of the files were annotated independently by all three annotators (IA). The rest of the files were annotated via the PA procedure. Periodical agreement meetings were held during and after both phases. For six other connectives, *aslında* ‘actually’, *halde* ‘despite’, *nedeniyle* ‘because of’, *nedenle* ‘for this reason’, *ötürü* ‘due to’ and *yüzden* ‘so, because of this’, only PA annotations were carried out.

Table 1 provides the averaged pair-wise averaged inter-annotator agreement, i.e. annotator against annotator agreement, Kappa (K) coefficient values of the IA phase for the first group, where three independent annotators created the annotations independently.

Table 2 shows the K values of the second phase for the same group, where the PA procedure followed the agreement meetings of the independent annotations.

Connective	**Arg1**	**Arg2**
ama	0.832	0.901
sonra	0.820	0.902
ve	0.692	0.791
ya da	0.843	0.974

Table 1 – Pair-wise averaged inter-annotator agreement (K) for 3 individual annotators in IA – individual annotator against individual annotator

Connective	**Arg1**	**Arg2**
ama	0.956	0.969
sonra	0.889	0.953
ve	0.945	0.964
ya da	0.939	0.973

Table 2 – Inter-annotator agreement (K) for pair vs. individual in PA – individual annotator against pair annotator

In tables 1 and 2, all the cells but one, indicate good agreement ($0.80<K<1.00$). Only the first argument of *ve* ‘and’ in independent annotation phase shows not good but some agreement ($0.60 < K < 0.80$). Zeyrek et al. (2010) discusses other connectives in TDB with K values below 0.80.

The results show that the K values for both arguments have increased after the transition from the IA to PA. A repeated measures test shows that the increase is significant ($p< 0.01$).

Tables 3 and 4 show the agreement statistics for the second group of connectives, where only PA was conducted. Each set of annotations are compared to the agreed annotations that were produced after the final agreement meeting for that particular connective. In Table 3, the K values show the agreement between the individual’s annotations and the agreed annotations, and in Table 4, they indicate the agreement between the pair’s annotations and the agreed annotations.

Connective	**Arg1**	**Arg2**
aslında	0.766	0.889
halde	0.834	0.898
nedeniyle	0.905	0.984
nedenle	0.952	0.987
ötürü	1.000	0.907
yüzden	0.916	0.983

Table 3 - Individual annotator vs. agreed agreement (K) in PA

Connective	Arg1	Arg2
aslında	0.937	0.984
halde	0.973	1.000
nedeniyle	0.937	0.984
nedenle	1.000	1.000
ötürü	1.000	0.953
yüzden	0.992	1.000

Table 4 – Pair annotator vs. agreed agreement (K) in PA

In Tables 3 and 4, except for the mediocre agreement of Arg1 of *aslında* ‘actually’, all K values indicate good agreement. A repeated measures test shows that the agreement of the pair and the agreed annotations are significantly higher than the agreement of the individual annotator and the agreed annotations ($p<0.001$).

Since non-discourse connectives were omitted during the annotation phase instead of being marked as non-discourse connectives, there was no easy way to distinguish errors from deliberate omissions in type (d) divergences. In an attempt to find out the missing annotations, we compared the number of relations that were annotated both on the agreed annotations and on the annotators’ annotations. At first glance it seems that the individual annotator missed significantly more relations that should be annotated than the pair ($p<0.5$). However, since many of the cases omitted by the individual annotator were of type (e) divergences similar to (9), this comparison does not yield interpretable results.

5 Discussion and Conclusion

For the first group of connectives discussed in this paper, where a 3-annotator independent annotation procedure preceded the PA procedure, there was a significant increase in the K values for inter-annotator agreement, which probably due to the agreement meetings that took place between the two annotation phases. In the agreement meetings, peculiar uses of specific connectives and syntactic structures unique to the connectives were explored. Following the discussions, some annotation guidelines were added, modified or fine-tuned, new principles were added or modified to reflect the annotators’ intuitions both about general properties of Turkish discourse structure and the particular discourse connective in question.

As a result, annotators were prepared for the PA phase, leading to less disagreement between the individual annotator and the pair.

For the second group of connectives, where all annotations were carried out with one individual and a pair, the higher K values for the pair vs. agreed annotations than for the individual vs. agreed annotations reflect the benefits of pair programming.

During PA, simple mistakes are corrected during annotation. Ambiguities are discovered more easily because the annotators discover different readings and point them to each other, and discuss productively in an attempt to agree on the more prominent reading. Annotation principles are applied more carefully because the pair is usually more alert than the individual. PA allows for better understanding and analysis of the context, because the sum of the contextual and world knowledge of the partners is greater than that of the individual annotators. As a result, the annotation is more accurate and although not statistically proven yet, it is observed to be faster.

5.1 Conclusion

The benefits of the PA can be summarized as higher annotation clarity due to less annotation errors and faster disagreement resolution due to previous extended discussions. The drawbacks are one less set of annotations for each pair of annotators and the shadow of doubt cast over the unbiased nature of annotations due to the dense interaction of the pair. While pair jelling was beneficial for PP, it might prove problematic for PA, as independent linguistic intuition is valuable in linguistic annotation. We believe that we have minimized this bias by treating the pair as a single annotator for the agreement statistics, and by letting the individual intuitions and ideas leak into the agreement meeting by means of the notes field in the annotation tool. However, this solution was project specific and the problem should be investigated in more detail when applying PA to other projects.

Acknowledgements

We gratefully acknowledge the support of the Turkish Scientific and Technological Research Council of Turkey (TUBITAK) and METU Scientific Research Fund (no. BAP-07-04-2011-005).

References

Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse.* Kluwer Academic Publishers.

Berfin Aktaş, Cem Bozşahin, Deniz Zeyrek. 2010. Discourse Relation Configurations in Turkish and an Annotation Environment. *LAW IV - The Fourth Linguistic Annotation Workshop.* Uppsala, Sweden, July 2010.

Kent Back. 2000. *Extreme Programming Explained: Embrass Change*. Addison Wesley Longman, Reading, Mass.

Frederick, P. J. Brooks. 1975. *The Mythical Man-Month*. Addison-Wesley, Reading, Mass.

Tom DeMarco and Timothy Lister. 1977. *Peopleware*. Dorset House, New York.

Eleni Miltsakaki, Rashmi Prasad, Aravind Joshi, Bonnie Webber. 2004. Annotating Discourse Connectives and Their Arguments. *HLT/NAACL Workshop on Frontiers in Corpus Annotation*. Boston, MA. May 2004.

Beata Beigman Klebanov and Eyal Beigman. 2008. From Annotator Agreement to Noise Models. *Computational Linguistics*, 34(3):495–503.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, Bonnie Webber. 2008. The Penn Discourse Treebank 2.0. *LREC'08 - The sixth international conference on Language Resources and Evaluation.* Marrakech, Morocco, May 2008.

Dennis Reidsma and Jean Carletta. 2008. Reliability measurement without limit. *Computational Linguistics*, 34(3):319–326.

Bilge Say and Deniz Zeyrek and Kemal Oflazer and Umut Ozge. 2002. Development of a Corpus and a Treebank for Present-day Written Turkish. *ICTL 2002 - 11th International Conference on Turkish Linguistics.* Famagusta, TRNC, August 2002.

Laurie Williams, Robert R. Kessler, Ward Cunningham, Ron Jeffries. 2000. Strengthening the Case for Pair Programming. *IEEE Software*, July/August 2000:19–25.

Laurie Williams and Robert R. Kessler. 2000. All I Really Need to Know about Pair Programming I Learned In Kindergarten, *Communications of the ACM*, 43(5):108–114.

Laurie Williams and Robert R. Kessler. 2003. *Pair Programming Illuminated.* Addison Wesley, Reading, Massachusetts.

Deniz Zeyrek, Işın Demirşahin, Ayışığı Sevdik-Çallı, Hale Ögel Balaban, İhsan Yalçınkaya and Ümit Deniz Turan. 2010. The Annotation Scheme of the Turkish Discourse Bank and an Evaluation of Inconsistent Annotations. *LAW IV - The Fourth Linguistic Annotation Workshop* Uppsala, Sweden, July 2010.