

Reichenbach ve Allen Zaman Mantığı ile TimeML

Şadi Evren ŞEKER¹, Banu DİRİ¹

¹ Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü, İstanbul

ab@sadievrenseker.com ,

banu@ce.yildiz.edu.tr

Özet: Bu çalışma kapsamında Allen zaman mantığını (AZM) kullanarak TimeML işaretleme dilinin nasıl metinlerdeki zamanları işaretlediği açıklanmıştır. Ayrıca, TimeML işaretleme dilinin, metinlerde bulunan olayların zamanlarını nasıl sıraladığı ve sıralanan bu olaylar üzerinde nasıl işlem yapıldığı incelenmiştir. Son olarak XML üzerine inşa edilen TimeML dili kullanılarak İnternet üzerindeki kaynakların zamansal analizlerinin yapılması, sıralanması ve zaman bilgisine göre aratılması imkanı tartışılmıştır.

Anahtar Sözcükler: Allen Zamansal Mantığı, TimeML.

Allen Temporal Logic and TimeML

Abstract: In this study, deployment of Allen's temporal logic in TimeML tagging language is studied. Also event ordering on natural language texts and post process on ordered events by TimeML is discussed. Finally, using event ordering of Internet resources and temporal analysis for searching via TimeML language built on XML is researched.

Keywords: Allen's Temporal Logic, TimeML.

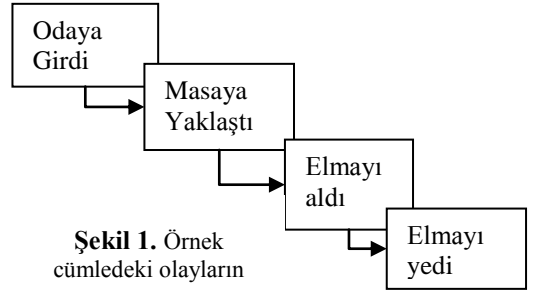
1. Zaman İşaretlemesi

Zaman işaretleme dilleri, doğal dilde bulunan metinler üzerinde kullanılan ve bu metinlerin içeriği hakkında makinelerin işleyebileceği gösterimlerin çıkarılmasını amaçlayan dillerdir. Zaman işaretleme dilleri esas olarak metinlerde bulunan zamansal varlıklara yoğunlaşır ve olayların ve durumların zamansal bağlantısını kurmayı hedefler. Bu konuda yapılan çalışmalar, internet arama motorlarında, soru cevaplama, metin özetleme veya metnin görselleştirilmesi gibi alanlarda kullanılabilir. [1][2]

Günümüzde zaman işaretleme dilleri özellikle doğrusal mantık üzerine kurulu olan Allen zaman mantığı [3] veya Reichenbach zamansal mantığı [4] gibi mantıklar üzerine inşa edilmiştir.

2. Allen Zamansal Mantığı

Allen zamansal mantığı (AZM), olaylar arasındaki bağlantıları ve olayların sıraları üzerinde çalışır. Örneğin “olay A, olay B’den öncedir” veya “olay A ile olay B aynı zamandadır” gibi işlemleri kullanarak olaylar arasındaki ilişkileri modellemeye çalışır.



Şekil 1. Örnek cümledeki olayların sırası

AZM üzerindeki en temel işlemler, iki parametresi bulunan ve iki olay arasındaki ilişkiyi betimlemeye yarayan işlemlerdir.

Bu durum aşağıdaki cümle üzerinden daha rahat anlaşılabilir:

“Ali, odaya girdikten sonra, masanın üzerindeki elmayı yedi”

Yukarıdaki bu cümledeki olaylar sıralanacak olursa şekil 1 deki gibi bir tablo ortaya çıkacaktır.

Şayet cümleler üzerinde daha detaylı bir işleme yapılırsa, Ali'nin ilk başta odanın dışında olduğunu biliyoruz, ayrıca odaya

girdikten sonra masanın üzerindeki elmayı alması için, masaya yaklaşması gerektiği de çıkarılabilir. Son olarak Ali'nin masadan elmayı alana kadar, aslında elinde elma olmadığı da doğrudur. Bu durumlar modele eklenecek olursa, şekil 2'de gösterilen tablo ortaya çıkar.

Yukarıdaki cümleden örneğin Ali'nin odanın dışındayken elması olmadığı veya Ali'nin masanın yanındayken bir elması olduğu veya Ali'nin elmayı yerken odada olduğu ve masanın yanında olduğu sonuçları çıkarılabilir.

Yukarıdaki olaylar zincirini modellemek için AZM üzerinde bazı değişkenleri tanımlayalım. Örneğin girme (G) eyleminin, odanın dışında (OD) olma durumunu gerektirdiğini ve odaya girme eyleminden sonra odanın içinde (Oİ) olma durumuna geçildiğini söyleyebiliriz. Benzer şekilde masaya yaklaşma (MY) eylemi, masadan uzakta olma (MU) durumunu masanın yakınında olma (MYO) durumuna çevirmiştir. Ayrıca elmayı alma (EA) eylemi de elması yok (EY) durumunu elması var (EV) durumuna çevirmiştir. Bütün bu eylemler aslında yemek (Y) eyleminin ön koşullarıdır.

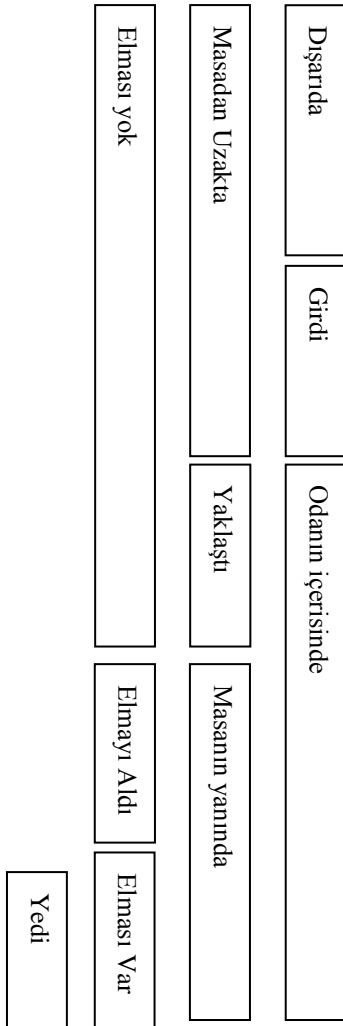
Yukarıdaki bu durum ve şartları aşağıdaki şekilde bağlamak mümkündür.

$$\begin{aligned} & \text{Meets}(\text{OD}, \text{G}) \wedge \text{Meets}(\text{G}, \text{Oİ}) \wedge \text{During}(\text{G}, \text{MU}) \wedge \text{During}(\text{MU}, \text{Oİ}) \wedge \\ & \text{Meets}(\text{MU}, \text{MY}) \wedge \text{Meets}(\text{MY}, \text{MYO}) \wedge \text{During}(\text{MU}, \text{EY}) \wedge \text{During}(\text{EA}, \text{Oİ}) \wedge \text{During}(\text{EA}, \text{MYO}) \wedge \\ & \text{Meets}(\text{EA}, \text{EA}) \wedge \text{Meets}(\text{EA}, \text{EV}) \wedge \text{During}(\text{Y}, \text{EV}) \wedge \text{During}(\text{Y}, \text{MYO}) \wedge \text{During}(\text{Y}, \text{Oİ}) \wedge \text{Meets}(\text{EA}, \text{Y}) \end{aligned}$$

Yukarıdaki model, çıkarılabilecek bazı zamansal ihtimalleri modellemektedir. Bunun yanında bir okuyucu, örnek cümlemizden, Ali'nin henüz elmayı almadan aslında aç olduğu sonucunu da çıkarabilir. Bu durumda modelimiz aşağıdaki şekilde olmalıdır:

$$\begin{aligned} & \text{Occurs}(\text{aç}, \text{EY}) \wedge \text{Holds}(\text{aç}, \text{EA}) \wedge \\ & \text{Meets}(\text{aç}, \text{Y}) \end{aligned}$$

Yukarıdaki modele göre, Ali'nin henüz elması yokken, Ali açtı, ve Ali elmayı



Şekil 2. Olaylar ve durumlar

aldığında da açtı ve Alinin aç olma durumu Ali'nin elmayı yemesine kadar sürdü anlamı çıkarılabilir.

Yukarıdaki bu işlemleri de kapsayan AZM işlemleri aşağıdaki şekilde sıralanabilir:

- Before (x,y) or After (y,x) {önce veya sonra}
- Overlaps (x,y) or Overlapped (y,x) {üst üste binmek}
- Meets(x,y) or MetBy(y,x) {bir olayın bitmesi ile başkasının başlaması}
- Contains(x,y) or During (x,y) {bir olayın bir diğerini zamansal olarak kapsaması}
- Starts(x,y) or StartedBy(y,x) {bir olayın bir diğerini başlatması}
- Ends(x,y) or EndedBy(y,x) {bir olayın bir diğerini bitirmesi}
- Equals(x,y) {iki olayın tam olarak aynı zamanda olması}

3. Reichenbach Zamansal Mantığı

Reichenbach zaman mantığı ise, metinlerdeki zamansal modellemeyi üç ana unsur üzerine kurar:

1. olayın geçtiği zaman (E)
2. olayın yazıldığı veya anlatıldığı zaman (S)
3. referans zamanı (R)

Örneğin aşağıdaki cümleyi ele alalım:

“Ali gelip gerçekleri anlattığında çoktan talimat vermişim.”

Yukarıdaki cümlede geçen fiilleri ayıklayacak olursak:

(“(Ali gelip gerçekleri anlattığında)R (çoktan talimat vermişim)E.”)S

Yukarıda gösterilmek istenen bütün cümlelerin şu anda anlatıldığı veya okunduğudur. Ayrıca talimat verme eylemi tam olarak olayın geçtiği zamanda gerçekleşmiştir. Son olarak Alinin gelip gerçekleri anlatması eylemi olayın zamanına referans alınan bir olaydır. Yani bu cümlede bu iki olay arasında bir bağlantı kurulmuştur.

Dolayısıyla Reichenbach analizine göre yukarıdaki cümlede 3 farklı zamanın sıralaması aşağıdaki şekilde olacaktır:

$E < R < S$

Yani olayın geçtiği zaman, referans zamanından çok önce olmuştur, benzer şekilde olayın referans zamanı ise olayın anlatıldığı şimdiki zamandan öncedir.

Yukarıdaki örnekten yola çıkarak Reichenbach 13 farklı zaman dizilimi alternatifi getirmiştir. Bu dizilimler aşağıdaki gibi bir tabloda toparlanabilir.

İlişki	Reichenbach Zaman İsmi (Tense Name)	İngilizce Zaman İsmi	Türkçe Karşılığı
$E < R < S$	Anterior past	Past perfect	Hikaye Geçmiş Zaman
$E = R < S$	Simple past	Simple past	Geçmiş Zaman
$R < E < S$			
$R < S = E$	Posterior past		Gelecek Zamanın Mazisi
$R < S < E$			
$E < S = R$	Anterior present	Present perfect	Rivayet Geçmiş Zaman
$S = R = E$	Simple present	Simple present	Geniş Zaman
$S = R < E$	Posterior present	Simple future	YOK
$S < E < R$			YOK
$S = E < R$	Anterior future	Future perfect	YOK
$E < S < R$			
$S < R = E$	Simple future	Simple future	Gelecek Zaman
$S < R < E$	Posterior future		

Yukarıdaki tabloda 13 farklı zaman gösterilmektedir. Bu zamanların hepsinin İngilizce veya Türkçe için birebir karşılıkları yoktur ancak konuşmacı (yada yazar) farklı şekillerde bu zamanları ifade etmektedir.

4. TimeML

TimeML, olaylara bağılı zaman kavramlarını ve bu zamanlar ve olaylar arasındaki ilişkileri tutmak için ağırlıklı olarak James Pustejovsky tarafından 2003 yılından beri geliştirilen XML tabanlı bir işaretleme dilidir. Dilin web üzerindeki sayfasına <http://www.timeml.org> adresinden erişilebilir. Dilde temel olarak 3 seviye öge bulunmaktadır bunlar:

1. Zaman içerikli olayların saklandığı ve ağacın yapraklarını oluşturan olaylar

2. Yaprakların üzerindeki ve zaman olayları üzerindeki, zaman belirleyici değerler. Bu değerlere dilde sinyal (işaret) anlamı verilmiştir.

3. Olaylar arasındaki bağlantılar (link)

TimeML dilindeki 1. ve 2. seviye olan, olaylar ve bu olayların zaman belirleyicileri Reichenbach zamansal mantığından esinlenilerek geliştirilmiştir. Benzer şekilde TimeML'in en üst seviyesini oluşturan ve olayları birbirine bağlamakta kullanılan olaylar arası bağlantı seviyesi Allen zamansal mantığından esinlenilerek hazırlanmıştır.

4.1. Zamansal Kayıtlar (yaprak seviyesi):

<EVENT>: En alt seviyedeki etiketlerdir ve olayları tutmak için kullanılır. Genel olarak her olay bir fiildir ve DTD kaydı aşağıda verilmiştir:

```
attributes ::= eid class
eid ::= ID
{eid ::= EventID
EventID ::= e<integer>}
class ::= 'OCCURRENCE' |
'PERCEPTION'
| 'REPORTING' | 'ASPECTUAL'
| 'STATE' | 'I_STATE' |
'I_ACTION'
```

Yukarıdaki tanımdan da anlaşılacağı üzere bir olayın ayırt edici bir ID değeri bulunmalı ve bu değer bir tam sayı olmalıdır. Ayrıca bir olay yukarıda tanımlanan sınıflardan (Class) bir tanesine dahil olabilir.

4.2. Olaylar arası bağlantı seviyesi

TimeML için olaylar arası bağlantıların modellenmesi Allen zamansal mantığından

esinlenerek hazırlanmıştır. AZM için tanımlı olan ilişki yapısı aşağıda verilmiştir:

```
lid ::= LinkID
eventInstanceID ::=
EventInstanceID
timeID ::= TimeID
signalID ::= SignalID
relatedToEventInstance ::=
EventInstanceID
relatedToTime ::= TimeID
relType ::= 'BEFORE' | 'AFTER'
| 'INCLUDES' | 'IS_INCLUDED' |
'DURING' | 'DURING_INV' |
'SIMULTANEOUS' | 'IAFTER' |
'IBEFORE' |
'IDENTITY' | 'BEGINS' | 'ENDS'
| 'BEGUN_BY' | 'ENDED_BY'
```

İki olay arasındaki bağlantının kurulmasında yukarıdaki yapıya uygun bir tanım gerekmekte ve ayrıca ilişki tipi AZM için tanımlı olan tiplerden (relType) olmalıdır. Bu ilişki tipleri tablo 2'de örnekleri ile birlikte verilmiştir:

relType	Sample
BEFORE	Geldi ve beni gördü.
AFTER	
INCLUDES	Ali, ankaraya geçen Perşembe vardı. .
IS_INCLUDED	
DURING	Ali, iki yıllığına öğrenci oldu.
DURING_INV	
SIMULTANEOUS	
IAFTER	Uçak dağa çarpınca bütün yolcular öldü.
IBEFORE	
IDENTITY	Ali bursaya arabası ile giderken köfte yedi.
BEGINS	Ali, 1980'den beri ders veriyor.
ENDS	Ali, saat 7'ye kadar okuldaydı.

BEGUN_BY	
ENDED_BY	

4.3. Örnek TimeML dosyası

TimeML dilinin örnek bir cümle için nasıl çalıştığı bu bölümde açıklanacaktır. Örnek cümlemiz, “Masanın yanına gidip elmayı alırken Ayşe uyandı” olsun. Bu cümlemin TimeML işaretlenmiş hali aşağıdadır:

```
Masanın yanına <EVENT eid="e1"
class="OCCURRENCE">gidip</EVENT>
elmayı <EVENT eid="e2"
class="OCCURRENCE">alırken</EVENT>
Ayşe
<EVENT eid="e3"
class="OCCURRENCE">uyandı</EVENT>.
```

```
<MAKEINSTANCE eventID="e1"
eid="ei1" tense="PAST"
aspect="NONE" polarity="POS"
pos="UNKNOWN"/>
<MAKEINSTANCE eventID="e2"
eid="ei2" tense="PAST"
aspect="NONE" polarity="POS"
pos="UNKNOWN"/>
<MAKEINSTANCE eventID="e3"
eid="ei3" tense="Hikaye"
aspect="NONE" polarity="POS"
pos="UNKNOWN"/>
```

```
<TLINK lid="l2"
relType="BEFORE"
eventInstanceID="ei1"
relatedToEventInstance="ei2"
origin="USER"/>
<TLINK lid="l1"
relType="IS_INCLUDED"
eventInstanceID="ei3"
relatedToEventInstance="ei2"
origin="USER"/>
```

Görüldüğü üzere cümlede sadece olay ve olaylar arasındaki bağlantıları belirten kelimeler işaretlenmiş, bunun dışındaki kelimeler serbest olarak bırakılmıştır. Ayrıca dosyanın sonunda olaylar arasındaki ilişkileri gösteren TLINK etiketleri bulunmaktadır.

4.4. TimeML ve Türkçe

TimeML yapısı itibarıyla Reichenbach ve Allen zaman mantıkları üzerine kurulu bir işaretleme dilidir. Bu makalede de anlatılmış olan bu iki zaman mantığı ne yazık ki Latin dil ailesi için üretilmiş olup, Türkçede ve Türkçe ile aynı zamansal mantığa sahip Türk dil ailesindeki dilleri tam olarak desteklememektedir. Örneğin Türkçede bulunan öğrenilen geçmiş zaman, Reichenbach zaman mantığında tam karşılığını bulamamaktadır. Benzer şekilde Türkçede bulunan ve bazı olay belirten fiillerin tekrarlanmasından oluşan ifadeler de Allen zaman mantığında gösterilememektedir. Örneğin aşağıdaki cümleyi ele alalım:

“Arabanın sinyali, yanıp sönyordu”

Bu cümlede iki ayrı fiil bulunur. Yanmak ve sönmek fiilleri arasındaki ilişki ise ne yazık ki Allen zaman mantığı tarafından modellenemez. Bunun sebebi tekrarlı olayların Türkçede birbirinin tersi fiillerle gösterilmesine karşılık Latin dil grubunda benzer bir yaklaşımın bulunmamasıdır. Örneğin İngilizcede bu tip ifadeler tek bir fiil ile karşılır.

5. Sonuç

Bu makalede, zamansal gösterimler için kullanılan ve varlıkbilim (ontoloji) veya doğal dil işleme çalışmaları için kullanılabilir olan TimeML işaretleme dili ve bu dile temel teşkil eden Reichenbach ve Allen zamansal mantıkları açıklanmış ve Türkçe için kullanılmasındaki güçlükler örneklenmiştir.

TimeML dilinin çıktısı olan XML sayesinde olay sıralama, kronoloji çıkarımı, soru cevaplama gibi alanlarda kullanılmasının mümkün olduğu gösterilmiştir.

6. Kaynaklar

- [1] Automatic Time Expression Labeling for English and Chinese Text, Kadri Hacioglu, Ying Chen and Benjamin Douglas Springer LNCS, Computational Linguistics and Intelligent Text Processing, ISSN 0302-9743 , pages 548-559 , 2005
- [2] Robust temporal processing of news, Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, Inderjeet Mani , George Wilson Pages: 69 - 76 , 2000
- [3] Allen Linear (Interval) Temporal Logic –Translation to LTL and Monitor Synthesis– Grigore Ro,sul _ and Saddek Bensalem, CAV'06, LNCS 4144, pp 263-277, 2006
- [4] Reichenbach, H., Elements of Symbolic Logic, New York: Macmillan (1947)
- [5] Ozkirimli A, Türk Dili: Dil ve Anlatım, İstanbul Bilgi University Press, 2001.
- [6] TimeML, TERQAS, (2002), TimeML has been developed in the context of three workshops starting from 2002.
- [7] “TimeML Annotation Guidelines Version 1.2.1” Roser Saur’ı, Jessica Littman, Bob Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky (January 31, 2006)
- [8] Natalia Kotsyba Using Petri nets for temporal information visualization Études Cognitives/Studia Kognitywne, CEEOL (2006)
- [9] Andr’e Bittar , Annotation of Events and Temporal Expressions in French Texts Proceedings of the Third Linguistic Annotation Workshop, ACL-IJCNLP 2009, pages 48–51, Suntec, Singapore, 6-7 August 2009. c 2009 ACL and AFNLP
- [10] Christian Kissig and Laura Rimell , Closing TLink-Relations (Reasoning with Intervals) June 23, 2005