Bilgisayar Ortamında Bir Derlem Geliştirme Çalışması

Bilge Say,
Enformatik Enstitüsü
Bilişsel Bilimler Ana Bilim Dalı
Orta Doğu Teknik Üniversitesi
bsay@ii.metu.edu.tr

Umut Özge,
Enformatik Enstitüsü
Bilişsel Bilimler Ana Bilim Dalı
Orta Doğu Teknik Üniversitesi
umut@ii.metu.edu.tr

Kemal Oflazer,
Mühendislik ve Doğa Bilimleri Fakültesi
Sabancı Üniversitesi
oflazer@sabanciuniv.edu

Öz: Bu bildiride dilbilim ve bilgisayarlı doğal dil işleme çalışmalarına kaynak olmak üzere elektronik ortama geçirilen günümüz Türkçesini yansıtan metin örneklemlerinin işaretlenmesiyle oluşturulan bir derlemin geliştirilme süreci özetlenmektedir. Özellikle, derlemin tasarımı ve kullanımı açısından gerçekleştirilen bilişim süreçlerinin detayları derlem oluşturma sürecinin genel perspektifi içinde verilecektir.

Ana Konu: Araştırmada bilişim.
Anahtar Sözcükler: Derlem, bilgisayarlı dilbilim, TEI, XML, XCES.

1. Akademik Bir Kaynak Olarak Derlem

Derlem (bütünce, corpus (İng.)) belli prensipler çerçevesinde özel veya genel amaçlı metin ya da konuşma parça ya da bütünlerinin, üzerinde yapılacak araştırmaya uygun işaretlemelerle beraber biraraya getirilmesinden oluşan bütündür. Günümüz derlemlerinin elektronik ortamda tutularak, erişim ve kullanım kolaylığı sağlanması yaygındır (Kennedy,1998). Bu da derlem oluşturma çabalarının bir akademik bilişim aktivitesi olarak ele alınmasını gerektirir.

Bir derlemden nasıl yararlanılabileceğine kısaca değinmek bu emek yoğun sürecin değerlendirilmesinde yararlı olacaktır. Dilbilimin pek çok altsahasında bir derlemden yararlanmak bilimsel bir metod olarak araştırmalara katkıda bulunmaktadır: bir sözlük oluşturulmasında günümüzün yeni sözcüklerinin nasıl bağlamlarda kullanıldığını araştırmak; sözdizimsel bir kuramın savlarını yaşayan dilin kesitlerinden örneklerle ve istatistiklerle desteklemek; bir dil öğrenimi sınıfında dilin yapılarına ve sözcüklerine örnekler vermek bu kullanımların sadece bazılarıdır (McEnery ve diğerleri, 1996). Doğal dil işleme alanında son yıllarda ağırlıklı olarak kullanılan istatistiki modellerin başarılı kullanımı için bol miktarda veriye, yani bir derleme ihtiyaç duyulur (Manning ve Schütze, 1999).

Özellikle Batı dillerinde son on-onbeş yılda tüm araştırmacıların kullanımına açık derlem

oluşturma çalışmaları meyvelerini vermiş; İngiliz Ulusal Derlemi (BNC, 2001), Çek Ulusal Derlemi (Cermák, 1997) gibi 100 milyon sözcükle hem sözel hem yazılı dili temsil eden kapsamlı derlemler oluşturulmuştur. Her ne kadar Türkçe üzerine çalışan dilbilimciler arasında yaptığımız bir ankette, katılanların %41'i Türkçe bir derlem kullandıklarını belirtmiş olsalar da bu derlemlerin çoğu kişisel olarak biraraya getirilmiş, küçük ölçekli, elektronik olmayan ve tüm akademisyenlerin ulaşımına açık olmayan derlemlerdir. Oysa hem Türkçe dilbilim çalışmalarının ve hem de Türkçe üzerine yapılan doğal dil işleme çalışmalarının (Oflazer 1997) gelişimi değişik türleri içeren, elektronik ortamda, telif haklarına saygılı, tüm araştırmacılara açık ve bilgisayar ortamında sorgu yazılımlarıyla desteklenen birden çok derleme ihtiyaç duyulduğunu göstermektedir.

2. ODTÜ Türkçe Derlemi Tasarımı

Bu bölümde hem ODTÜ Türkçe derlemi tasarımının genel öğeleri, hem de bu derlemin bir altderlemi olan ODTÜ-Sabancı ağaç yapılı derleminin genel hatları verilecektir.

2.1 Genel Tasarım Öğeleri

Bu gereksinmeden yola çıkan ODTÜ Türkçe Derlemi Geliştirme projesi kısıtlı bütçe ve personel olanaklarıyla önceki bölümde bahsedilen gereksinimlere belirli sınırlamalar çerçevesinde yanıt olacaktır. Bu sınırları kısaca gözden geçirelim: Derlem sadece yazılı dili içermekte, daha yoğun emek gerektirdiğinden sözlü dil kapsam dışı tutulmaktadır. Türk dilinin gelişiminden çok günümüzdeki durumunu temsil etmeyi amaçlayarak 1990 sonrasında yayınlanan eserlerden örneklemler alınmaktadır. Derlem dilin değişik yönlerinin incelenmesi açısından değişik türlerde (Bkz. Bölüm 3) örneklemler içermekte, ancak bu türlerin derlem içerisindeki yüzdesi dilin üretimi ve tüketimine dair bir araştırmaya dayanmamaktadır (Biber, 1993). Büyüklük açısından 2 milyon sözcük hedeflenmekte; bu Brown derlemi gibi 80'lerde sık kullanılan 1 milyon sözcüklük derlemlere göre büyük, ancak sözlükbilimsel çalışmalar için alt sınır kabul edilen 10 milyon sözcük gereksinimine göre küçük bir miktardır (Kennedy, 1998). Burada da sınırlayıcı etken eldeki iş gücü olanağının kısıtlılığı olmuştur. Bir örneklemdeki sözcük sayısı Brown ve benzerlerinde olduğu gibi 2000'dir ki bu seçimde de belirleyici etken, yayınevlerinin genelde uzun örneklemlere telif hakları açısından sıcak bakmaması olmuştur. Derlemin bir kısmı biçimbirimsel ve sözdizimsel işaretlenmek üzere ayrıca projelendirilmiştir (Bkz. Bölüm 2.2). Derlemin tümü dünyada metin işaretleme standardı olarak kabul gören TEI (Text Encoding Initiative) standardının derlemlere özel bir XML uygulaması olan XCES'le işaretlenmektedir (Bkz. Bölüm 3).[1]

Bu tasarım parametrelerinin belirlenmesiyle başlayan derlem oluşturma çalışmalarında derlem oluşturma sürecinde iki ana bilişimsel aktivite ortaya çıkmıştır: Metin örneklerinin elektronik ortama alınması ve işaretlenmesi ile dilbilimcilerin kolay kullanımı için genel amaçlı bir sorgu yazılımı geliştirilmesi. Bölüm 3'de bu süreçler detaylı olarak tanıtılacaktır.

2.2 ODTÜ-Sabancı Ağaç Yapılı Derlemi Tasarımı

Penn Ağaç Yapılı Derlemi (Penn) gibi derlemler son yıllarda teorik dilbilim ve doğal dil işleme çalışmalarında, geliştirme ve geçerliliği ölçme açısından önemli kaynaklar olmuşlardır. Bu açıdan Türkçe derlemin bir kısmının biçimbirimsel ve sözdizimsel olarak işaretlenmesi Sabancı Üniversitesi ile işbirliği içinde ayrı bir proje olarak gerçekleştirilmektedir. Türkçenin biçimbirimsel (morfolojik) olarak zengin bir

[1] Derlem tasarımı ile ilgili detaylı bilgi için bkz. Atkins ve diğerleri (1993).

dil olması, tümcenin öğeleri arasında sözcüklere eklenen sonlu bir işaret kümesine dayanan değil, "çekimsel grup" adını verdiğimiz biçimbirimsel altyapılara dayanan bağlılık (dependency (İng.)) yapılarının işaretlenmesinin dilbilimsel ve bilişimsel açıdan anlamlı bir gösterim olduğu sonucunu getirmiştir. (Oflazer ve diğerleri, 2000). Bu sonuçtan yola çıkan bir tasarımla şu ana kadar bir işaretleme kılavuzu ve işaretlemenin kolaylıkla gerçekleştirilmesi için bir yazılım geliştirilmiştir. Bundan sonra biçimbirimsel olarak ayrıştırılan öğelerin çoklu yapılarının indirgenmesi (disambiguation (İng)), öbek yapısı gösteren "şırıl şırıl" gibi dizilerin işaretlenmesi ve kalan öğeler arasında özne, nesne, belirteç gibi fonksiyonel bağların yarı otomatik işaretlenmesi planlanmaktadır.

3. Derlem Yapım Süreci

Bu bölümde derlemin yapım aşamasında yer alan süreçlere değineceğiz. Derlem yapımı, *metin toplama, işaretleme* ve *kontrol* olmak üzere üç ana süreci kapsamaktadır. Takip eden bölümlerde bu süreçleri ayrı ayrı ele alacak, süreçlerin detayları, uygulanan standartlar ve kullanılan yazılımlar hakkında bilgi vermeye çalışacak, bu kısmın son bölümünde de derlemi sorgulamak amacına yönelik, henüz yapım aşamasındaki derlem sorgu yazılımının özelliklerine değineceğiz.

3.1 Metin Toplama

Metin toplama süreci, derleme alınacak eserlerin telif hakkı sahiplerinden yazılı izin alınması ile başlayıp, metinlerin bir sonraki süreç olan işaretleme süreci öncesi elektronik ortamda "unicode salt metin" formatında hazır bulunmasıyla sona erer.

Derleme alınacak metinlerin belirlenmesinde rol oynayan faktörlerden biri, derlem içeriğinin türlere göre dağılımının dengeli olmasıdır. Bu dağılımı takip edebilmek amacıyla 14 metin türü kategorisi belirlenmiştir. Bu makalenin yazıldığı sırada 1.000.000 sözcüğe (2000'er kelimelik 250 örneklem) ulaşan derlem içeriğinin türlere göre dağılımı Tablo 1'deki gibidir.

Tür	**Dağılım (%)**
Roman	24
Öykü	21
Makale	16
Deneme	14
Araştırma-İnceleme	12
Gezi	4
Söyleşi	2
Diğer	7

Tablo 1: Derlem içeriğinin türlere göre dağılımı.

Yukarıdaki tablodaki "Diğer" başlığı *köşe yazısı, referans, anı, yaşam öyküsü, özyaşam öyküsü, kişisel gelişim* ve *ders kitabı* türlerini kapsamaktadır.

Derleme dahil edilecek eserler, tercihen elektronik ortamda yayıncıdan sağlanmaya çalışılmıştır. Bunun mümkün olmadığı durumlarda üniversite kütüphanelerinden veya yayıncı kuruluştan temin edilmektedir. Bu eserlerin içinden seçilen örneklemler HP ScanJet 6200C tarayıcı ile taranıp, OCR tekniğiyle "salt metin" formatında elektronik ortama aktarılmaktadır. Tarama işlemi sırasında bazı

karakterlerin yanlış tanınması sonucu oluşan hatalar elle düzeltilmekte, bu yolla elektronik ortama aktarma işlemi sırasında orijinal metne sadık kalınmaktadır. 2000 kelimeden oluşan bir örneklemin kaynağından taranarak elektronik ortama aktarılması bir çalışanın 1 ½ saatini almaktadır.

3.2 İşaretleme

Derlemimizde bulunan elektronik metinlerin kodlanmasında, metinlerin basılı bulundukları kaynaktan bağımsız olarak paragraf, tırnak, listeleme ve benzeri öğelerinin ve ayrıca künye bilgilerinin kodlanması için oluşturulan TEI (Text Encoding Initiative) uygulaması olan XCES'i kullanmaktayız.

1980'lerin sonuna doğru internet'in de yaygınlaşmaya başlamasıyla, sayıca artan elektronik metinlerde bulunan bilginin, donanım, yazılım ve uygulamalardan bağımsız bir şekilde paylaşılması ve verimli bir şekilde işlenilmesi sorunu doğmuştur. Bu sorunu çözmek için bir grup araştırmacının başlattığı 7 yıl süren çalışma 1994 yılında ilk resmi TEI kılavuzunun yayınlanmasıyla meyvesini vermiştir (Sperberg-McQueen ve Burnard, 1994). TEI, metinlerin elektronik ortamda gösterimine evrensel bir standart getirmeyi amaç edinmiş uluslararası akademik bir araştırma hareketidir.

TEI bizlere elektronik metinlerin kodlanmasına yönelik çok temel düzeyde SGML işaretlerinden ve ortaya çıkacak SGML dokümanlarının düzgün yapılandırılmış olma koşullarını belirleyen DTD (Document Type Definition)'lerden oluşan bir standart sunmaktadır. Bu standardın eldeki amaca uygun şekilde özelleştirilerek geliştirilmesi gerekmektedir. Dilbilimsel derlemler için TEI kılavuzları ile uyumlu bir SGML (Standard Generalized Markup Language) uygulaması olan ve dilbilimsel derlemlerin genel mimarisini ve dilbilimsel işaretleme standartlarını belirleyen CES (Corpus Encoding Standard) elektronik metin kodlama kılavuzu geliştirilmiştir (Ide, 1996).

Şu ana kadar bahsettigimiz metin kodlama standartlarının hepsi SGML uygulamalarıdır. Günümüzde web üzerindeki veriyi görüntülemekte neredeyse evrensel bir dil olarak kabul görmüş HTML'in de tabanını oluşturan SGML, ilk başta yukarıda bahsettigimiz standardizasyon için en uygun dil olarak görülüyordu. Fakat çok geçmeden SGML'in bir takım eksileri göze çarpmaya başladı. Bu eksiler: (i) SGML'in son derece karmaşık bir sözdiziminin olması; (ii) SGML dokümanlarının DTD olmadan ayrıştırılamaması; (iii) SGML dokümanlarının kısmen ayrıştırılma şansının olmaması ve bunun özellikle büyük dokümanlarda zaman kaybına yol açması olarak özetlenebilir (DeRose, 1999). Bu noktadan hareketle, yukarıda özetledigimiz eksilerden arındırılmış, SGML'in gücü ve esnekliği ile HTML'in basitliğini birleştiren XML dili geliştirilmiştir (Bray ve diğerleri, 1998).

Biz de bu bağlamda CES'in, bilişim ve internet teknolojisinde hızla yaygınlaşan XML diline adapte edilmiş hali olan XCES'i derlemimizde işaretleme standardı olarak kullanmayı uygun bulduk. CES'in XML'e adapte edilmesi halen devam eden bir süreçtir. (Welty ve Ide, 1999, s.62).

Derlemi oluşturan ve her biri 2000 sözcüklük bir metin ve bu metni işaretlemekte kullanılan XCES işaretlerini içeren bir XML dokümanı olan örneklemler, *başlık* (header) ve *gövde* (body) olmak üzere iki bölümden oluşmaktadır.

Başlık bölümünde örneklemin alındığı kaynağın detaylı bibliyografik bilgilerinin yanı sıra, örneklemin yazılı olduğu dosyanın adı, dosyanın büyüklüğü, örneklemin içerdiği sözcük sayısı (işaretler hariç), kontrol işlemi (Bkz. Bölüm 3.3) sırasında eğer metnin orjinali üzerinde bir değişiklik yapıldıysa, ne tür bir değişikliğin, kim tarafından ne zaman yapıldığı gibi bilgiler yer almaktadır. Şekil 1'de tipik bir örneklem başlığı görülmektedir.

```
<cesHeader>
 <fileDesc>
  <titleStmt>
   <h.title>00017113</h.title>
  </titleStmt>
  <extent>
    <wordCount>2008</wordCount>
    <byteCount>17929</byteCount>
  </extent>
```

```
  <sourceDesc>
   <biblStruct>
    <analytic>
     <h.title>Anadolu Dağlarının 'Bitki Avcısı': Prof. Dr. Turhan BAYTOP</h.title>
     <h.author>Nalân MAHSERECİ</h.author>
    </analytic>
    <monogr>
     <h.title></h.title>
     <h.author></h.author>
     <edition></edition>
     <imprint>
       <publisher>Bilim ve Ütopya</publisher>
       <pubDate>Mart 2000</pubDate>
       <pubPlace>İstanbul</pubPlace>
     </imprint>
     <idno>1301 - 6717</idno>
     <biblScope>69</biblScope>
    </monogr>
   </biblStruct>
  </sourceDesc>
 </fileDesc>
 <profileDesc>
  <textClass>
   <catRef>Makale</catRef>
  </textClass>
 </profileDesc>
 <revisionDesc>
  <change>
   <changeDate>12.10.2000</changeDate>
   <respname>Sedef</respname>
   <h.item>The header part was changed.</h.item>
  </change>
 </revisionDesc>
</cesHeader>
```

Şekil 1: Tipik bir örneklem başlığı.

Örneklemin *gövde* bölümünde üç aşamalı bir standart olan CES'in asgari işaretlemeyi belirleyen birinci aşaması uygulanmıştır (Ide, 1996, Bölüm 4.1). *Gövde* kısmında kullandığımız işaretlere kısaca göz atalım.

(a) Üst paragraf düzeyi işaretlerinin bazıları:

<text> metinleri işaretler.
<body> metin içinde bütünlüğü olan parçaları işaretler. (örneklemin ana kısımlarında biri olan *gövde* (body) kısmıyla karıştırılmamalıdır.)
<opener> metinlerin başlangıcındaki, tarih, anahtar sözcükler ve benzerlerini işaretler.
<head> metin, liste şiir gibi yapıların başlıklarını işaretler.

(b) Paragraf Düzeyi İşaretlerinin bazıları:

<p> paragrafları işaretler.
<q> tırnak içine alınmış kısımları işaretler.
<hi> normal karakter formatı dışında yatık, koyu, altıçizili gibi vurgulanmış sözcükve sözcük öbeklerini işaretler.
<poem> şiirleri işaretler.
<table> tabloları işaretler.
<list> listeleri işaretler.
<note> metin içinde geçen her türlü notu işaretler.
<abbr> kısaltmaları işaretler.

<date> tarihleri işaretler

Şekil 2'de Nihal Yeğinobalı'nın Can Yayınları tarafından basılmış "Sitem" adlı romanından alınmış bir örneklemin *gövde* kısmının kısaltılmış hali görülmektedir.

```
<text>
<body>
.
.
<p>Oktay biraz önce, <q>Hadi biz de Sitem'in yanına gidelim,</q> demişti. Sitem'in, kucağında
Tomurcuk Beyle Yılanlı İncirlerden yana gittiğini o da görmüştü çünkü. Ben omuz silkmekle
yetindim, Oktay da üstelemedi. Sitem ikimizin yüzüne karşı da görünmez kapılar kapamıştı. Benim
de elinden kayıp gidivermemden korkan Oktay beni <hi>oyalamak</hi> için geçen yaz Giray
Ağabeysiyle Kirazlı Yaylaya yaptıkları bir gezintiyi anlatmaya başladı.</p>
.
.
<p>O gün ve sonrasında olanları elbet sana da anlatmışlardır, Dalya. Gene de o kargaşa, o
şaşkınlık, o panik, o kafa karmaşası yaşanmadan bilinemez...</p>
.
.
</body>
</text>
```

Şekil 2: Tipik bir örneklem gövdesi.

İşaretleme süreci, elektronik ortama aktarılmış ham metnin, grubumuzca C'de yazılmış, en sık kullanılan “<p>” ve “<q>” işaretlerini otomatik olarak metne yerleştiren bir yazılımla işlenmesiyle başlar. Bu safhadan sonraki işaretleme işlemlerinde yine grubumuzca Borland C++ Builder'da yazılmış XCESEdit adını verdiğimiz bir XML editörü kullanılmaktadır. Grafik kullanıcı arayüzüne sahip bu program örneklemin *gövde* kısmında işaretleyicinin imlecin bulunduğu yerlere sağ fare tuşunu kullanarak gerekli işaretleri koyabilmesine olanak sağlamaktadır. İşaretleyici hiçbir işareti kendisi yazmadığı için işaretleri yanlış yazması olasılığı ortadan kalkmaktadır. *Gövde* kısmının işaretlenmesinin tamamlanmasının ardından, editörün *başlık* oluşturma işlevi kullanılır. *Başlık* oluşturma işleminde editör işaretleyicinin önüne, alanları Şekil 1'deki gibi bir *başlığın* işaretlerini kapsayan bir form getirir. Bu form doldurulurken işaretleyici herhangi bir alana uygun olmayan bir bilgi girerse yazılım tarafından uyarılır. (Ör. örneklemi oluşturan metnin türünün belirten <catRef> alanına projede kullanılan türler dışında herhangi birşey girerse yazılım hata mesajı verir). Form doldurulduktan sonra örneklemin *başlığı* otomatik olarak oluşturulur. Tüm bu işlemler bittikten sonra yine aynı editör kullanılarak örneklem ayrıştırılır. Editör, açılan işaretlerin kapanıp kapanmadığını, işaretlerin içiçe geçip geçmediğini kontrol ederek hata olan yerleri işaretleyiciye bildirir. Böylece düzgün yapılandırılmış bir XML dokümanı oluşturulmuş olur. Bahsedilen yazılımlar kullanılmaya başlanmadan önce bir işaretleyicinin yaklaşık 1 saat 15 dakikasını alan işaretleme süreci yazılımların kullanılmasıyla 1/2 saatte tamamlanabilmektedir.

3.3 Kontrol

Yazılım desteğine rağmen gözden kaçan bir takım hatalar olabilmektedir. Bu nedenle bir önceki bölümde anlattığımız şekilde oluşturulan örneklemler son bir defa daha kontrol edilmektedir. Kontrolü yapan kişi bu safhada örneklemi orijinal metinle karşılaştırır. Orijinal metinde imla hatası yapıldığından şüphelendiği durumlarda Türk Dil Kurumu’nun 2000 yılı baskısı "İmla ve Yazım Kılavuzu"'na başvurarak gerekli düzeltmeyi yapar ve bu düzeltmeyi de metin içinde kodlar.

Her örnekleme, aynı zamanda örneklemin yazılı olduğu dosyanın da ismi olan 8 haneli bir kod numarası verilmektedir. Kod numarasının ilk 5 hanesi örneklemin alındığı kaynağı belirtmektedir. Kaynaklar derleme dahil edilme sıralarına göre 00001’den başlayarak numaralandırılmıştır. İlk beş haneyi takip eden hane ise o örneklemin eldeki kaynaktan alınan kaçıncı örneklem olduğunu belirtmektedir. Bir sonraki hane örneklemi hangi çalışanın işaretlediğini göstermektedir. (Proje çalışanları tek basamaklı bir

sayıyla kimliklendirilmiştir.) Sekizinci ve son hane ise örneklemin kontrolünü yapan kişiyi belirtmeye ayrılmıştır. Örneklemlerin kodlanmasını bir örnek ile açıklayacak olursak; 00125273 kodlu bir örneklem "125" no'lu kaynaktan alınan ikinci örneklem olup "7" kimlik numaralı çalışan tarafından işaretlenip "3" kimlik numaralı işaretleyici tarafından kontrol edilmiştir. Bu kodlama sistemi ve *başlık* kısmında kodlanan değişiklik bilgileri sayesinde hataların kimler tarafından yapıldığı takip edilebilmekte, çalışanların yaptıkları hatalar konusunda uyarılmasıyla derlemin kalitesi yüksek düzeyde tutulmaktadır.

3.4 Derlem Sorgu Yazılımı Tasarımı

Derlem, kullanıcılara Java dilinde yazılmış, kullanıcıya grafik bir arayüz kullanarak derlemi sorgulama imkanı sağlıyacak bir derlem sorgu yazılımı ile birlikte dağıtılacaktır. Bu bölümde henüz yapım aşamasında olan bu yazılımın bazı özelliklerine değineceğiz.

Derlem sorgu yazılımının geliştirilmesinde göz önünde bulundurduğumuz iki ana tasarım kriteri şunlardır;

(i) Kullanıcıların derlemden elde etmeyi amaçladıkları bilgi türündeki çeşitliliği hesaba katarak, mümkün olduğu kadar geniş bir yelpazede sorgu yapabilme imkanı tanımak.

(ii) Sorgular karşılığında derlemden çıkarılan bilginin kolay analiz edilmesini sağlayacak, bir taraftan başlangıç düzeyindeki bilgisayar kullanıcılarının bile zorluk çekmeden kullanabileceği kadar basit, diğer taraftan da ileri düzey kullanıcılara kullanım esnekliği sağlayan bir grafik arayüz sunabilmek.

Derlem sorgu yazılımı istemci/sunucu mimarisinde, nesne tabanlı yaklaşımla Java programlama dilinde yazılmaktadır. Java dilinin seçilmesinin nedeni, sözkonusu programlama dilinin değişik platformlarda büyük ölçüde sorunsuz çalışabilen, unicode karakter kümesi kullanması itibariyle Türkçe karakterlerde sorun çıkarmayan ve web üzerinde çalışabilen uygulamalar yaratılmasına olanak sağlamasıdır.

Kullanıcıların derlemden sorgulamak ihtiyacını duyacağını düşündüğümüz ve sorgulama yazılımının olanak vereceği sorgular şöyledir;

1. derleme göz atma;
2. sözcük sorgusu;
3. XCES işaretlerine göre sorgulama;
4. bağlılık yapısına göre sorgulama (ağaç yapılı derlem);
5. biçimbirimsel özelliklere göre sorgulama;
6. düz terim (regular expression(İng.)) sorgulama;
7. çapraz sorgular yapabilme.

Kullanıcı, bu sorgular karşısında elde ettiği sonuçları tarih ve saatiyle birlikte kaydedebilecektir. Kullanıcı, yazar adı, yayınevi, tür, basım yılı vb. kriterler belirterek yalnızca bu kriterlere uyan örneklemleri kapsayan bir alt derlem oluşturabilecek, ve sorgularını bu alt derlem üzerinde yapabilecektir. (ör. 1995-1996 yılları arasında basılmış, öykü türündeki metinlerdeki alıntıları sorgulama.) Ayrıca kullanıcı bu alt derlemleri kaydedebilme, dolayısıyla istediği zaman tekrar kullanabilme imkanına da sahip olacaktır. Yazılım, prototip olarak tamamlandığında bir grup dilbilimci tarafından denenecek ve gerekli düzeltmeler yapılacaktır.

4. Sonuç

Türk Dili açısından bir ilk olacak olan bu derlem projesi, Mayıs 2002'de ağaç yapılı altderlem kısmı dışında sona erecektir. Yaşadığımız deneyimin bu bildiriyle aktarılmasının daha kapsamlı ya da özel amaçlı başka derlemlerin gerçekleştirilmesine katkıda bulunacağını umuyoruz.

Teşekkür

Projede halen veya başlangıçta fikren ve/veya emeği ile katkısı olan öğretim üyeleri Prof. Dr. Wolf König, Prof. Dr. Deniz Zeyrek, Doç. Dr. Cem Bozşahin, Y. Doç. Dr. Ümit Deniz Turan, Y. Doç. Dr. Margaret Sönmez, Dr. Ayşenur Birtürk, Dr. Dilek Hakkani-Tür ve Dr. Gökhan Tür'e; araştırma görevlileri Barış Şükrü Demiral, Barış Çağrı Genç, ve Filiz Yılmaz Bican'a; ve "işaretleyicilerimiz" Sedef Akgül, Aygün Boduroğlu, Deniz Cantürk, Devrim Saran ve Barış Şara'ya; bizi maddi olarak destekleyen ODTÜ AFP (AFP No: 99-06-04-02) ve TÜBİTAK'a (EEEAG Proje No: 199E026); metin toplamada bize yardımcı olan tüm yayınevi ve kuruluşlara (Can Yayınları, İletişim Yayınevi, Bilgi Yayınevi, Kuraldışı Yayınevi, Adam Yayınları, İşbankası Kültür Yayınları, Yapı Kredi Yayınları, Bilim ve Ütopya, Doğu-Batı, Atlas ve Bütün Dünya dergileri.); ve Türkçemize "derlem" sözcüğünü kazandıran ve deneyimlerini bizimle paylaşan Prof. Dr. Aydın Köksal'a teşekkür ederiz.

Kaynakça

Atkins, S, J. Clear, ve N. Ostler. 1993. *Corpus Design Criteria.* Literary and Linguistic Computing 8(4).

Biber, D. 1993. *Representativeness in Corpus Design.* Literary and Linguistic Computing 8(4).

BNC. British National Corpus. http://info.ox.ac.uk/bnc/, Aralık 2001 haliyle.

Bray, T., J. Paoli, ve C. M. Sperberg-McQueen. 1998. *Extensible Markup Language (XML) 1.0. W3C Recommendation*, Şubat.

DeRose, S.J. 1999. *XML and the TEI.* Computers and the Humanities 33, 11-30.

Cermák, F. 1997. *Czech National Corpus: A Case in Many Contexts.* International Journal of Corpus Linguistics 2(2).

Ide N. 1996. *Corpus Encoding Standard: Document CES 1, Sürüm 1.4,* Ekim. http://www.cs.vassar.edu/CES/

Kennedy, G. 1998. *An Introduction to Corpus Linguistics.* Longman.

Manning, C. D. ve Schütze. H. 1999 *Foundations of Statistical Natural Language Processing.* Cambridge MA: MIT Press.

Mc Enery, T ve A. Wilson. 1996. *Corpus Linguistics.* Edinburgh: Edinburgh University Press.

Oflazer, K. 1997. *Natural Language Processing Research in Turkey.* Proceedings of 3rd Telri European Seminar, Montecatini, Italy, Oct 16-18.

Oflazer. K, B. Say, D. Z. Hakkani-Tür, G. Tür. 2000. Building a Turkish Treebank. Abeille A (haz.) *Building and Exploiting Syntactically Annotated Corpora.*

Penn Treebank. http://www.ldc.upenn.edu/, Aralık 2001 haliyle.

Sperberg-McQueen, C. M., ve L. Burnard, (haz.) 1994. *Guidelines for Electronic Text Encoding and Interchange (TEI P3),* Chicago, Oxford: Text Encoding Initiative.

Welty, C., N. Ide. 1999. *Using the Right Tools: Enhancing Retrieval from Marked-up Documents.* Computers and the Humanities 33, 59-84.