

PEMANFAATAN BAHASA ALAMI SEBAGAI ANTAR MUKA APLIKASI NL-FARAID

Bayu Setiaji¹, Ema Utami², Hanif Al Fatta³

¹Mahasiswa Magister Teknik Informatika, Program Pasca Sarjana STMIK AMIKOM Yogyakarta

^{2,3}Dosen Magister Teknik Informatika, Program Pasca Sarjana STMIK AMIKOM Yogyakarta

Jl. Ring Road Utara, Condong Catur, Sleman, Yogyakarta

Telp: (0274) 884201-207, Fax: (0274) 884208 Kodepos: 55283

email: bayusetiaji@amikom.ac.id¹, ema.u@amikom.ac.id², hanif.a@amikom.ac.id³

ABSTRACT

Patrimony science or faraid is a branch of science in Islamic law that discusses the ins and outs of the division of the estate based on descent in the family hierarchy. All the rules of calculation derived from the Qur'an and are discussed in the book devoted to the science of inheritance. It's now widely available software that can be used as a tool for the calculation of the distribution of the estate so that it can be used by a layman though. The software is currently still used as a form-like based interface.

Natural Language Processing or NLP is a branch of Artificial Intelligence or AI that allows the computers to understand natural language similar to those used by humans. While still on the use of written language, NLP has allowed the interaction between humans and computers can be done with natural language, no longer use the mediation form or the like. With the use of NLP can be made an heir by the name of counter application NL-Faraid (Natural-language Faraid) using natural language as a medium of inter-face.

The results showed this application can be used to calculate the distribution of the estate. Applications can accept input and produce output calculations in the Indonesian language.

Keywords: natural language processing, natural language, Indonesian language, faraid

1. PENDAHULUAN

Menurut *Buku II Kompilasi Hukum Islam (KHI)* hukum kewarisan atau *faraid* adalah hukum yang mengatur tentang pemindahan hak pemilikan harta peninggalan (tirkah) pewaris, menentukan siapa-siapa yang berhak menjadi ahli waris dan berapa bagiannya masing-masing. Dalam buku tersebut dijelaskan beberapa hal yang berhubungan dengan hukum waris, yaitu:

- a. Pewaris adalah orang yang pada saat meninggalnya atau yang dinyatakan meninggal berdasarkan putusan Pengadilan beragama Islam, meninggalkan ahli waris dan harta peninggalan.
- b. Ahli waris adalah orang yang pada saat meninggal dunia mempunyai hubungan darah atau hubungan perkawinan dengan pewaris, beragama Islam dan tidak terhalang karena hukum untuk menjadi ahli waris.
- c. Harta peninggalan adalah harta yang ditinggalkan oleh pewaris baik yang berupa benda yang menjadi miliknya maupun hak-haknya.

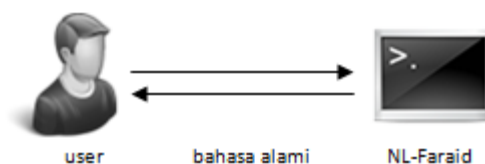
- d. Harta waris adalah harta bawaan ditambah bagian dari harta bersama setelah digunakan untuk keperluan pewaris selama sakit sampai meninggalnya, biaya pengurusan jenazah (tajhiz), pembayaran hutang dan pemberian untuk kerabat.

Fokus pemrosesan dari NLP sendiri sebenarnya adalah kalimat. Kalimat dapat dirumuskan sebagai sintaksis terbesar yang terdiri dari dua kata atau lebih. Hubungan struktural antara kata dan kata, atau kelompok kata dan kelompok kata lain, berbeda – beda. Antara kalimat dan kata terdapat dua satuan sintaksis antara, yaitu klausa dan frasa. Klausa merupakan satuan sintaksis yang terdiri dari dua kata atau lebih, yang mengandung unsur predikasi. Unsur predikasi adalah subjek, predikat, objek, pelengkap, atau keterangan. Frasa adalah satuan sintaksis yang terdiri dari dua kata atau lebih yang tidak mengandung unsur predikasi (Krullee, 1991) dalam (Wiryan, 2001) dalam (Desiani, 2006).

Berkaitan dengan aplikasi NL-Faraid ini maka penggunaan NLP di sini adalah sebagai *translator* bahasa alami ke bahasa buatan yang dipahami atau dieksekusi oleh aplikasi. Contohnya adalah “hitunglah pembagian harta waris bila seseorang meninggal dengan harta 100 juta meninggalkan seorang istri dan 2 orang anak lelaki!”. Kemudian aplikasi akan menjalankan perintah tersebut dan menampilkan hasilnya dalam bahasa alami.

2. NL-FARAIID

NL-Faraid adalah aplikasi penghitung waris berbasis konsol yang input maupun outputnya menggunakan perantara bahasa alami. Bahasa yang digunakan sebagai perantara adalah bahasa Indonesia. Input dapat berupa kalimat utuh maupun kata – kata kunci yang disusun sebagai frase – frase. Secara umum proses input output aplikasi NL-Faraid adalah sebagai berikut:



Gambar 1. Input Output NL-Faraid

Inti dari aplikasi ini adalah fungsi yang memuat logika perhitungan waris yang bersumber dari pasal – pasal dalam Buku II KHI. Selain itu aplikasi juga memiliki kata – kata kunci yang mencakup kelompok ahli waris dan harta waris. Aplikasi ini juga memiliki database sederhana yang menyimpan daftar sinonim dan pembetulan ejaan kata – kata kunci yang kemungkinan salah tulis.

Aplikasi NL-Faraid bersifat *single user* ditulis menggunakan bahasa PHP yang berbasis CLI (*command line interface*). Aplikasi ini juga membutuhkan database SQLite sebagai media penyimpanan.

Proses *parsing* yang dilakukan aplikasi ini menggunakan operasi *Perl-compatible regular expression* (PCRE). PCRE adalah satu set fungsi yang menerapkan pencocokan pola ekspresi reguler menggunakan sintaks dan semantik yang sama seperti Perl 5 (www.pcre.org).

Operasi *regular expression* dilakukan pada input untuk mendapatkan kata – kata kunci yang sudah didefinisikan dan mendapatkan nilai – nilai untuk tiap – tiap kata kunci. Hasil *parsing* akan disusun dalam bentuk *array* yang kemudian diproses oleh fungsi penghitung waris. Hasil perhitungan akan dirangkai kembali dalam bentuk kalimat. Bila aplikasi gagal menghitung maka akan menampilkan pesan gagal.

3. ATURAN PEMBAGIAN WARIS

Aturan pembagian waris diatur dalam *BAB III Buku II HKI* tentang Besarnya Bahagian. Aturan – aturan tersebut tertuang dalam pasal 176 sampai 191. Berikut ini adalah tabel pembagian harta waris:

Tabel 1. Pembagian harta waris

A	B	C
1	Bila tidak ada anak/cucu	1/4
	Bila ada anak/cucu	1/8
2	Bila tidak ada anak/cucu	1/2
	Bila ada anak/cucu	1/4
3	Sendirian (tidak ada anak dan cucu lain)	1/2
	Dua atau anak perempuan tidak ada anak atau cucu laki-laki	2/3
4	Sendirian atau bersama anak / cucu lain (laki-laki atau perempuan)	Ashobah (sisa seluruh harta setelah dibagi pembagian lain)
	Keterangan : Pembagian antara laki-laki dan perempuan 2 banding 1	
5	Bila tidak ada anak / cucu	1/3
	Bila ada anak / cucu	1/6
6	Bila tidak ada anak/cucu dan tidak ada dua saudara atau lebih dan tidak bersama Ayah Kandung	1/3
	Bila ada anak/cucu dan / atau ada dua saudara atau lebih dan tidak bersama Ayah Kandung	1/6
	Bila tidak ada anak/cucu dan tidak ada dua saudara atau lebih tetapi bersama Ayah Kandung	1/3 dari sisa sesudah diambil istri/janda atau suami/duda
7	Sendirian tidak ada anak / cucu dan tidak ada Ayah Kandung	1/6
	Dua orang lebih tidak ada anak / cucu dan tidak ada Ayah Kandung	1/3
8	Sendirian tidak ada anak / cucu dan tidak ada Ayah Kandung	1/2
	Dua orang lebih tidak ada anak / cucu dan tidak ada Ayah Kandung	2/3
9	Sendirian atau bersama saudara lain dan tidak ada anak / cucu DAN tidak ada ayah kandung	Ashobah (sisa seluruh harta setelah dibagi pembagian lain)
	Keterangan : Pembagian antara laki-laki dan perempuan 2 banding 1	
10	Menggantikan kedudukan orang tuanya yang menjadi ahli waris. Persyaratan berlaku sesuai kedudukan ahli waris yang diganti	Sesuai yang diganti kedudukannya sebagai ahli waris

Keterangan kolom:

A. Ahli Waris, yang terdiri dari:

- 1) Istri / janda
- 2) Suami / duda
- 3) Anak perempuan
- 4) Anak lelaki

- 5) Ayah kandung
- 6) Ibu kandung
- 7) Saudara seibu
- 8) Saudara perempuan kandung atau seayah
- 9) Saudara lelaki kandung atau seayah
- 10) Cucu / keponakan

B. Syarat

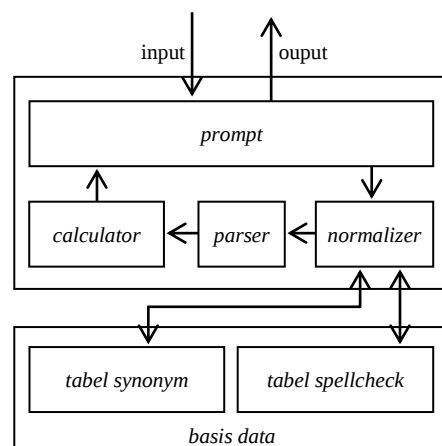
C. Besarnya Bahagian

Dalam pasal 174 dijelaskan bahwa bila semua ahli waris ada maka yang berhak mendapat warisan hanya anak, ayah, ibu, janda atau duda.

Seluruh aturan yang sudah didefinisikan di atas diimplementasikan dalam sebuah fungsi.

4. STRUKTUR APLIKASI

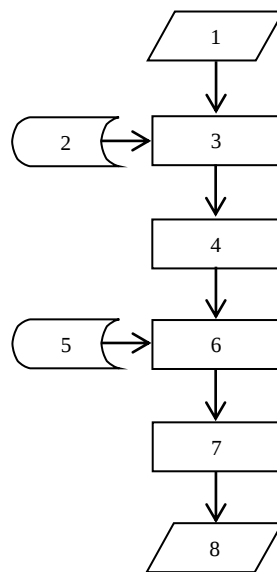
Aplikasi NL-Faraid terdiri dari beberapa bagian kecil yang masing – masing memiliki fungsi tersendiri. Secara umum struktur aplikasi NL-Faraid adalah sebagai berikut:



Gambar 2. Struktur aplikasi

Berikut ini adalah keterangan tiap – tiap bagian (modul) berdasar gambar 2 di atas:

1. *Prompt*, bertugas menerima input dan menampilkan output.
2. *Normalizer*, bertugas menormalkan input yang dikirimkan oleh interpreter. Normalisasi meliputi pembetulan ejaan yang salah, dan penggantian kata yang bersinonim dengan kata kunci. Hasil normalisasi adalah berupa *clean phrases* yang dapat diterima oleh *parser*. Berikut ini alur proses normalisasi:

**Gambar 2.** Flowchart normalisasi

Berikut ini adalah penjelasan proses pada gambar 2 di atas:

- 1) Input
- 2) Tabel *spellcheck*
- 3) Mengumpulkan pasangan kata salah (*misspelling*) dan kata pembetulan (*correction*)
- 4) Mengganti semua *misspelling* dalam input dengan *correction*
- 5) Tabel *synonym*
- 6) Mengumpulkan pasangan sinonim (*synonym*) dan kata kunci (*keyword*)
- 7) Mengganti semua kata yang memiliki sinonim dengan *keyword*
- 8) Hasil input normal

Hasil proses pada gambar 2 di atas diilustrasikan dalam tabel 1 berikut:

Tabel 2. Perbandingan input

Sebelum	Sesudah
kekayaan 100 jt	harta 100000000 1
seorang istri 2	orang istri 2
anak cowok dan 1	anak lelaki dan 1
anak cewe	anak perempuan

3. *Parser*, bertugas menerjemahkan susunan frase yang sudah dibentuk oleh normalizer menjadi *array* yang berisi *mapping* simbol dan nilai untuk kemudian diproses oleh

calculator. Ahli waris dan harta waris disimbolkan menjadi angka – angka. Berikut ini adalah simbolisasi berdasar keterangan tabel 1 sebelumnya:

Tabel 3. Simbolisasi

Simbol	Arti
0	Harta
1	Istri
2	Suami
3	Anak Perempuan
4	Anak Lelaki
5	Ayah
6	Ibu
7	Saudara Seibu
8	Saudara Perempuan Kandung atau Seayah
9	Saudara Laki – laki Kandung atau Seayah
10	Cucu

Proses *parsing* menggunakan operasi *regular expression*. Input akan diekstrak sehingga hanya diperoleh semua kata kunci dan semua bilangan kunci sebagai nilai untuk tiap kata kunci.

Parser akan mengelompokkan semua kata kunci dan mengelompokkan semua bilangan kunci yang ada. Berikut ini adalah ilustrasi prosesnya:

```
harta 100000000 dengan 1 istri 2 anak lelaki dan 1 anak
perempuan
```

Didapat:

- 1) Kata kunci: *harta, istri, anak lelaki, anak perempuan*
- 2) Bilangan kunci: *100000000, 1, 2, 1*

Dari kata kunci dan bilangan kunci tersebut maka berdasar urutan dapat dilakukan operasi penggabungan (*merging*) sehingga menghasilkan pemetaan antara kata kunci ke bilangan kunci. Berikut ini adalah bentuk pemetaannya:

```
harta → 100000000
istri → 1
anak lelaki → 2
anak perempuan → 1
```

Pemetaan di atas dapat diubah dalam bentuk pemetaan simbol:

```
{0:100000000;1:1;4:2;3:1}
```

Pemetaan simbol tersebut berbentuk *array* yang kemudian dikirim ke *calculator* untuk dihitung.

4. *Calculator*, bertugas menghitung pembagian waris. Bagian ini berupa fungsi yang berisi rangkaian struktur IF berisi logika pembagian harta waris berdasar pada tabel 1 sebelumnya. Hasil perhitungan dikirimkan kembali ke interpreter untuk diubah kembali ke bahasa alami.
5. Tabel *spellcheck*, berisi daftar kata salah beserta pembetulannya. Tabel ini digunakan oleh *normalizer* untuk membetulkan ejaan. Berikut ini adalah struktur tabel *spellcheck*:

Tabel 4. Struktur tabel *spellcheck*

Kolom	Tipe	Keterangan
id	int	Kode
misspelling	varchar	Kata yang salah
correction	varchar	Koreksi kesalahan

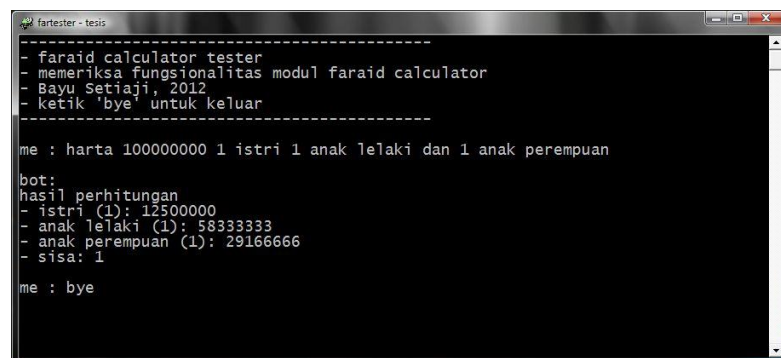
6. Tabel *synonym*, berisi daftar pemetaan kata – kata yang bersinonim dengan kata kunci yang sudah didefinisikan dalam aplikasi. Berikut ini adalah struktur tabel *synonym*:

Tabel 5. Struktur tabel *synonym*

Kolom	Tipe	Keterangan
id	int	Kode
synonym	varchar	Sinonim kata kunci
keyword	varchar	Kata kunci

5. PENGUJIAN

Pengujian dilakukan dengan memberikan beberapa *test case* berupa *frase* dan kalimat yang sudah dikenali kata kuncinya, dan hasil perhitungan juga berupa kalimat. Berikut ini adalah tampilan aplikasi NL-Faraid:



Gambar 3. Tampilan NL-Faraid

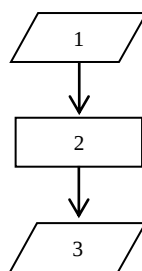
Berikut ini adalah tabel hasil pengujiannya:

Tabel 6. Hasil pengujian

No	Test Case	
1	input	harta 100000000 istri 1
	output	istri (1): 25000000 sisa: 75000000
	status	sukses
2	input	harta 100000000 anak perempuan 2
	output	anak perempuan (2) 50000000
	status	sukses
3	input	harta 100000000 anak perempuan 1
	output	anak perempuan(1): 100000000
	status	sukses
4	input	harta 100000000 anak lelaki 2 anak perempuan 1
	output	anak lelaki (2): 40000000 anak perempuan (1): 20000000
	status	Sukses
5	input	harta 100000000 istri 1 anak lelaki 1 anak perempuan 1
	output	istr i (1): 12500000000 anak lelaki (1): 58333333 anak perempuan (1): 29166666
	status	Sukses
6	input	harta 100000000 ayah kandung 1 ibu kandung 1 istri 1
	output	ayah kandung (1) 50000000 ibu kandung (1) 25000000 istri (1) 25000000
	status	sukses
7	input	harta 100000000 ayah kandung 1 ibu kandung 1 anak lelaki 1 anak perempuan 1
	output	ayah kandung (1): 16666666 ibu kandung (1): 16666666 anak lelaki (1): 44444444 anak perempuan (1): 22222222
	status	sukses
8	input	harta 100000000 anak lelaki 2 anak perempuan 2 istri 1 ayah kandung 1 ibu kandung 1
	output	anak lelaki (2): 18750000 anak perempuan (2): 9375000

		istri (1): 12500000 ayah kandung (1): 16666667 ibu kandung (1): 14583334
	status	sukses
9	input	harta 100000000 seorang istri
	output	istri (1): 25000000 sisa: 75000000
	status	sukses
10	input	seorang meninggal dgn kekayaan 100juta 2 orang anak cowok dan seorang anak cewek
	output	anak lelaki (2): 40000000 anak perempuan (1): 20000000
	status	sukses
11	input	seorang meninggal dgn duit 100juta 2 orang anak cowok dan seorang anak cewek
	output	anak lelaki (2): 0 anak perempuan (1): 0
	status	gagal
12	input	seseorang meninggal dengan harta 100000000 meninggalkan seorang istri 2 anak lelaki dan 1 anak perempuan. hitung pembagian harta warisnya
	output	istri: 0 anak lelaki(2): 0 anak perempuan(1): 0
	status	gagal

Pengujian menekankan pada proses ekstraksi input yang dilakukan oleh parser menjadi map *keyword* ke nilai. Gambar berikut ini menunjukkan proses ekstraksi input pada tiap *test case*:



Gambar 4. Ekstraksi

Keterangan:

- 1) Input
- 2) Ekstraksi input
- 3) Map *keyword* ke nilai

Test case nomor 1 sampai 8 menggunakan input berupa *keyword* yang sudah dikenali tepat berpasangan dengan nilainya sehingga parser dapat melakukan *mapping* dengan tepat. *Test case* nomor 9 dan 10 menggunakan input berupa sinonim untuk *keyword* yang berpasangan tepat

dengan nilainya. *Test case* nomor 9 ini sama artinya dengan nomor 1 dan nomor 10 sama artinya dengan nomor 4.

Sebagai contoh berikut ini adalah detail ekstraksi input pada *test case* nomor 4:

Input:

```
harta 1000000000 anak lelaki 2 anak perempuan 1
```

Ekstraksi:

```
Keyword: harta, anak lelaki, anak perempuan  
nilai: 1000000000, 2, 1
```

Map:

```
harta → 1000000000  
anak lelaki → 2  
anak perempuan → 1
```

Test case nomor 11 gagal karena ada kata yang tidak bersinonim dengan *keyword* yaitu *duit*. Ekstraksi input *test case* nomor 11 menghasilkan jumlah *keyword* dan nilai yang tidak sama sehingga *parser* salah melakukan mapping yang mengakibatkan *calculator* salah melakukan perhitungan.

Berikut ini adalah detail ekstraksi input pada *test case* nomor 4:

Input:

```
seorang meninggal dgn duit 100juta 2 orang anak cowok dan seorang  
anak cewek
```

Ekstraksi:

```
Keyword: anak lelaki, anak perempuan  
nilai: 1000000000, 2, 1
```

Map:

```
anak lelaki → 1000000000  
anak perempuan → 2  
null → 1
```

Test case nomor 12 gagal karena ada pengulangan *keyword* yaitu *harta*. Ekstraksi input *test case* nomor 11 menghasilkan jumlah *keyword* dan nilai yang tidak sama sehingga *parser* salah melakukan mapping yang mengakibatkan *calculator* salah melakukan perhitungan.

Berikut ini adalah detail ekstraksi input pada *test case* nomor 4:

Input:

seseorang meninggal dengan harta 100000000 meninggalkan seorang istri 2 anak lelaki dan 1 anak perempuan. hitung pembagian harta warisnya

Ekstraksi:

Keyword: harta, istri, anak lelaki, anak perempuan, harta
 nilai: 100000000, 1, 2, 1

Map:

harta → 100000000
 istri → 1
 anak lelaki → 2
 anak perempuan → 1
 harta → null

Ekstraksi test case nomor 12 menghasilkan 2 map dari *keyword* yang sama, yaitu *harta*, tetapi nilainya berbeda. Parser akan menganggap map yang terakhir yang digunakan sehingga *calculator* akan melakukan perhitungan yang salah.

6. KESIMPULAN

Dari hasil pengujian aplikasi NL-Faraid dapat disimpulkan bahwa:

- Aplikasi hanya dapat menghitung pembagian harta waris.
- Aplikasi hanya dapat menerima input berupa frase atau kalimat yang di dalamnya berisi kata – kata kunci atau kata – kata yang bersinonim dengan kata kunci.
- Proses perhitungan waris dapat dilakukan hanya bila banyaknya kata kunci sama dengan banyaknya nilai.
- Daftar sinonim untuk kata kunci dapat ditambahkan sebanyak mungkin dalam basis data.

7. SARAN

Berikut ini beberapa saran yang dapat diberikan untuk pengembangan aplikas NL-Faraid agar lebih baik:

- Dalam proses normalisasi dapat ditambahkan fungsi koreksi ulang terhadap kata kunci untuk menghindari duplikasi kata kunci saat akan di-*parsing*.
- Aplikasi ini dapat dilengkapi dengan fasilitas tanya jawab lain seputar istilah – istilah dalam faraid sehingga dapat juga digunakan sebagai sarana bantu belajar untuk pengguna.

8. PUSTAKA

Anonim. 2012. Kompilasi Hukum Islam. Bandung: Nuansa Aulia

Desiani, Anita, dkk., 2006. *Konsep Kecerdasan Buatan*. Yogyakarta: Penerbit Andi

<http://www.pcre.org> (tanggal akses 29 Oktober 2012)

<http://www.php.net/manual/en/book.pcre.php> (tanggal akses 29 Oktober 2012)