

Automatische Generierung von Übungsfällen

Michael Sonntag

Institut für Informationsverarbeitung und Mikroprozessortechnik (FIM), Johannes Kepler Universität Linz
Altenbergerstraße 69, A-4040 Linz
sonntag@fim.uni-linz.ac.at

Schlagworte: E-Learning, Fallgenerator, Ontologie

Abstract: Ein Problem des E-Learning ist, dass aufgrund eines vergrößerten Verhältnisses an Lernenden zu Lehrenden eine persönliche Betreuung schwieriger wird. Im juristischen Bereich wirkt sich dies u. a. so aus, dass von Lernenden selbstständig auszuarbeitende Fälle nicht immer persönlich korrigiert werden können bzw. manchmal solche Beispiele, zusammen mit ihren Lösungen, nicht in ausreichender Zahl zur Verfügung stehen. Zur Reduktion dieses Skalierungsproblems wird eine Software entwickelt, welche Übungsfälle für Lernende automatisch und adaptiv erzeugen kann. Damit ist es möglich, praxisnahe Übungsszenarien zur Verfügung zu stellen und autonomes und selbstgesteuertes Lernen zu ermöglichen.

1. Einleitung

Im juristischen Bereich hat E-Learning bereits Fuß gefasst¹, beruht jedoch vielfach auf einer 1:1-Übertragung von klassischem Unterricht, z. B. DVDs mit Videoaufzeichnungen von Vorlesungen oder Text-Chats. Im Geiste des Business Process Reengineering sollten jedoch zusätzlich die erweiterten bzw. neuen Möglichkeiten der Informations- und Kommunikationstechnik eingesetzt werden.

Während es für Lehrende bei Präsenzunterricht etwa praktisch unmöglich ist, persönlich Übungsfälle für jeden einzelnen Lernenden „maßzuschneidern“, ist eine derartige Anpassung durch computergestützte Generierung und Adaptivität bereits heute möglich. Die ersten Ergebnisse eines entsprechenden Forschungsprojektes mit Fokus auf Adaptivität und Zusammenarbeit von Lernenden werden hier vorgestellt. Thematisch

¹ Siehe <http://www.linzer.rechtsstudien.at/> als ein Beispiel.

besteht im hier vorgestellten System eine Fokussierung auf ein einzelnes Rechtsgebiet: Fälle zum materiellen Recht der UDRP (Uniform Domain-Name Dispute-Resolution Policy²) der ICANN. Dieses Gebiet ist zwar eng, doch von hoher Bedeutung für die Praxis: Jedes Jahr werden mehr als 2000 Domainstreitigkeiten nach diesem Schiedsverfahren entschieden, da es schneller und billiger als Gerichtsverfahren abläuft. Darüber hinaus bildete es auch die materielle Vorlage für eine Vielzahl ähnlicher Schiedsregeln bzw. -verfahren, u. a. dem für.eu Domains³, was seine praktische Bedeutung weiter steigert.

2. Erzeugung von Fall und Musterlösung

Die Generierung des Falles basiert darauf, Textfragmente aus jeweils ca. 1–3 Sätzen zu einem Fall (siehe das Beispiel in Abschnitt 4.1) zusammensetzen, wobei die Auswahl in der einfachsten Form nach dem Zufallsprinzip erfolgt. Querreferenzen, z. B. der Domainname bzw. der Name des Beschwerdeführers oder des Domaininhabers, werden entsprechend dem grammatikalischen Fall eingesetzt, wobei die jeweiligen „Slots“ in den Textbausteinen vordefiniert sind (siehe dazu Abschnitt 4.2). Schlussendlich wird die Groß-/Kleinschreibung korrigiert, was jedoch lediglich bei den Slots erforderlich ist. Hierbei wird auf eine einfache Heuristik zurückgegriffen, wonach beispielsweise ein Satzanfang immer groß geschrieben wird⁴.

Um sicherzustellen, dass der Ergebnisfall auch tatsächlich in rechtlicher wie realer Hinsicht „möglich“ ist, wurde eine Ontologie (siehe Abb. 1) aufgestellt. Hierbei werden praktisch wichtige Rechtsprobleme hierarchisch aufgegliedert. Für die Modellierung des Falles selbst sind entsprechende Felder vorhanden und es wurde die Klasse „UDRPCase“, welche einen Fall modelliert, mit einer Vielzahl an Einschränkungen versehen, um sicherzustellen, dass keine inkompatiblen Elemente zusammen ausgewählt werden. Hierdurch kann zwar nicht garantiert werden, dass generierte Fälle

2 <http://www.icann.org/udrp/> Für.com,.net und.org verbindlich.

3 Kapitel VI der VO (EG) Nr. 874/2004 der Kommission vom 28. April 2004, ABl. L 162/40 vom 30. 4. 2004 (<http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32004R0874:DE:HTML>).

4 Besondere Berücksichtigung erfordern hier Abkürzungen, welche ebenfalls das Satzzeichen „.“ enthalten können: So ist nach „z. B.“, „etc.“... keine Umwandlung auf Großschreibung vorzunehmen!

auch sinnvoll sind, aber es ist zumindest sichergestellt, dass es sich um rechtlich und sachlich mögliche Kombinationen handelt: Z. B. kann der Domaininhaber nicht gleichzeitig 100 € und 10.000 € für den Domainnamen verlangen; sehr wohl ist aber eine Kombination von 500 € und eines Ersatz-Domainnamens möglich.

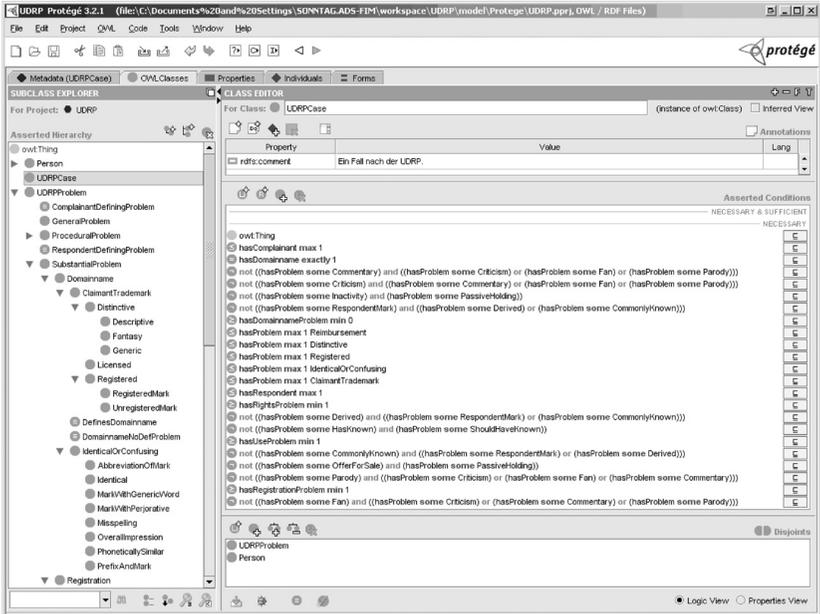


Abbildung 1: Darstellung eines Teils der Ontologie sowie der Regeln im Editor

Die Erzeugung des Falles erfolgt schrittweise: Es wird immer ein Problem hinzugefügt und anschließend festgestellt, ob es sich noch immer um einen konsistenten Fall handelt. Sofern nicht, wird der Textbaustein ausgeschieden und aus den verbleibenden ein neuer gewählt. Als Abschluss wird in einem weiteren Schritt geprüft, ob der Fall auch vollständig ist: Manche Probleme bedingen einen bestimmten Domainnamen oder einen speziellen Domaininhaber/Beschwerdeführer. Daher werden alle noch offenen Konfigurationselemente mit zufälligen Werten gefüllt. Hierbei erfolgt keine Konsistenzprüfung mehr, da keine Einschränkungen für nicht durch Textbausteine determinierte Elemente bestehen. Solche könnten jedoch bei Bedarf problemlos eingefügt werden, ebenso wie die dann auch hier erforderliche schrittweise Prüfung.

Zu berücksichtigen ist, dass Fälle entsprechend der UDRP rechtlich gesehen sehr einfach aufgebaut sind, da nur zwei Ergebnisse möglich sind: Der Domainname wird entweder vom Inhaber auf den Beschwerdeführer übertragen, oder dies erfolgt nicht. Das Ergebnis ist daher ein binäres. Ebenso ist die rechtliche Struktur der Fälle einfach: Es müssen kumulativ vier Elemente erfüllt werden (notwendig und hinreichend), sodass es zu einer Übertragung kommt. Diese sind:

- Der Beschwerdeführer besitzt eine Marke, zu welcher der strittige Domainname verwechslungsfähig ähnlich ist. Zum Beispiel: „Coca-Cola“ und www.cocakola.com (in der Ontologie durch die Klasse „Domainname“ und ihre Subklassen repräsentiert).
- Es muss böswillige Registrierung vorliegen, z. B. die Angabe falscher Daten im Whois oder Wissen von einer bevorstehenden Markenanmeldung. Die Modellierung erfolgt durch die Klasse „Registration“.
- Es muss böswillige Verwendung vorliegen, etwa ein Verkaufsangebot an den Markeninhaber für eine hohe Geldsumme oder der Versuch, durch die Verwechslung Benutzer anzulocken, z. B. durch die Einblendung von Werbebannern (siehe Klasse „Use“).
- Der Domaininhaber darf keine Rechte oder berechnigte Interessen an dem Domainnamen besitzen, wie sie u. a. eine eigene Marke oder die freie Meinungsäußerung darstellen (enthalten über die Klasse „Rights“).

Die Komplexität entsteht durch die Vielzahl der Varianten, durch welche diese Elemente erfüllt werden können. Diese vier Kategorien stellen gleichzeitig alle direkten Unterklassen der Klasse „SubstantialProblem“ dar, die inhaltliche Probleme beschreibt (siehe auch Abb. 1).

Aufgrund dessen ist die „Berechnung“ des Ausgangs des Falles fest codiert und basiert auf der Erfüllung dieser vier Elemente. Sollten komplexere Rechtsgebiete eingesetzt werden oder auch prozedurale Probleme integriert werden (verfahrensrechtliche Probleme sind nicht modelliert und werden nicht generiert bzw. behandelt), verkompliziert sich die Logik. In diesem Fall wäre eine Erweiterung um eine explizite Modellierung der Voraussetzungen bzw. der Verknüpfungen/Abhängigkeiten sinnvoll.

Ob eine Hauptkategorie durch den konkreten Sachverhalt erfüllt wird oder nicht, ist in der Ontologie angelegt: Jeder Textbaustein enthält einen Wert im Bereich von -100 bis $+100$, der angibt, zu welchem Ausmaß er alleine genügt, seine Kategorie zu erfüllen. Zur Entscheidung des Ergebnisses werden alle Werte der Instanzen eines konkreten Falles in dieser Kategorie aufsummiert und überprüft, ob die Summe 100 erreicht. Ein Wert von 0 eines Fragments bedeutet, dass es sich um einen „leeren“ Text handelt:

Dieser ist zur Ausschmückung und Verlängerung des Falles geeignet, aber ohne Einfluss auf das Ergebnis. Ein Wert von +100 bedeutet, dass dieser Textbaustein alleine für sich genügt, die Anforderung vollständig zu erfüllen, nachdem er etwa eine berechnete Nutzung oder eine böswillige Registrierung darstellt. Zwischenwerte sind nicht so eindeutig, sondern lediglich Hinweise in diese Richtung: Diesfalls sind mehrere Elemente erforderlich, um eine Erfüllung der Voraussetzung zu argumentieren. Negative Werte entsprechen den positiven, sind jedoch gegenläufige Argumente, ver- bzw. behindern also die Erfüllung des Merkmals. Um zu vermeiden, dass einzelne negative Merkmale eine bereits erfolgte Erfüllung (+100) zunichte machen, determiniert das Vorhandensein eines Einzelwertes von +100 oder -100 das Ergebnis absolut. Daher ist eine weitere Konsistenzbedingung des Falles, dass niemals diese beiden Werte gleichzeitig in derselben der vier Hauptkategorien eines Falles vorkommen dürfen.

Eine Limitierung des Ansatzes ist, dass aufgrund der zufälligen Generierung die Abfolge der Textbausteine nicht unbedingt einem einheitlichen logischen Aufbau folgt: So ist etwa eine Darstellung anhand des zeitlichen Ablaufs oder der Handlungen einer Person nicht möglich. Zusätzlich erfolgt eine Randomisierung der Reihenfolge, um zu verhindern, dass die Lernenden aus der bloßen Position eines Elementes auf seine Relevanz für eine bestimmte Hauptkategorie schließen können. In dieser Richtung ist eine Verbesserung auch unwahrscheinlich, da diesfalls eine Modellierung durch Textfragmente nicht mehr ausreichen würde, sondern eine echte Textgenerierung erforderlich wäre, welche dann jedoch zusätzlich ebenso einen argumentativen Aufbau berücksichtigen könnte.

3. Derzeitiger Stand der Implementierung

Der Fall-Generator ist in einer Grundversion fertig implementiert, wobei als Programmiersprache Java zum Einsatz kommt. Die Klassenhierarchie bzw. die Textfragmente werden mithilfe des Tools Protege⁵ editiert und sind in OWL repräsentiert, wobei die zu füllenden Felder innerhalb der Textfragmente, wie dem Domainnamen, ebenfalls (manuell) in XML modelliert sind. Für die Konsistenzprüfung sowohl im Editor wie auch bei der automatischen Generierung der Fälle wird der Reasoner Pellet⁶ eingesetzt. Eine

5 The Protégé Ontology Editor and Knowledge Acquisition System <http://protege.stanford.edu/>.

6 Pellet: The Open Source OWL DL Reasoner: <http://pellet.owldl.com/>.

konkrete Einschränkung ist derzeit, dass ein grafisches Benutzerinterface fehlt: Fall und Lösung werden nach der Generierung als Text ausgegeben (Näheres hierzu in Abschnitt 5).

4. Beispiel

Im Folgenden ist ein kurzer Beispielfall (nur die Angabe ohne die deutlich längere Musterlösung) dargestellt, welcher vollautomatisch generiert wurde.

4.1 Generierter Beispielfall

shoebox.com: DaimlerChrysler AG vs. Nuclear Marshmallows Inc.

Der strittige Domainname lautet „shoebox.com“, welchen DaimlerChrysler AG für sich reklamiert. Die Marke „ShoeBox“ ist zwar nicht für die DaimlerChrysler AG eingetragen, aber wurde über lange Zeit von der DaimlerChrysler AG benutzt und die DaimlerChrysler AG ist auch unter ihr allgemein bekannt. Die Nuclear Marshmallows Inc. behauptet keine Rechte an der Marke „ShoeBox“. Die Nuclear Marshmallows Inc. registrierte die Domain im Dezember 1994, worauf die DaimlerChrysler AG ein Streitbelegungsverfahren einleitete und der Name „on hold“ geschaltet wurde. Nach der Registrierung des Domainnamens durch die Nuclear Marshmallows Inc. erfolgte keine weitere Aktivität: Weder wurde eine Webseite darunter eingestellt, noch findet er eine sonstige Verwendung. Kurz nach der Registrierung wurde der Domainname an die Firma „Mail.com“ verkauft (dieser Vertrag wurde bis heute noch nicht erfüllt, da shoebox.com immer noch auf „hold“ steht!). Mail.com möchte die Adresse als „Vanity“-Adresse verwenden (z. B. „Michael.Sonntag@shoebox.com“). Die Nuclear Marshmallows Inc. verlangt, dass für 6 Monate auf der Startseite ein Link mit einem Hinweistext zur neuen Homepage eingerichtet wird.

4.2 Zwei Beispiels-Textfragmente

Bei diesen Fragmenten ist zu sehen, auf welche Weise die einzelnen variablen Elemente repräsentiert werden: Beispielsweise wird am Ende der Generierung „<Domainname />“ durch den konkreten Domainnamen ersetzt; analog dazu „<Mark />“ durch die Marke. Komplexer ist die Repräsentation des Beschwerdeführers durch „Complainant“: Hier finden sich zwei Parameter (technisch: Attribute des Elements), wobei der erste, der grammatikalische Fall, obligatorisch ist. Dies ermöglicht es, passend zur

(ebenso per Zufall ausgewählten) konkreten Entität den korrekten Text einfügen: „der Acme AG“ bzw. „dem Herrn Müller“. Der zweite Parameter legt fest, dass in diesem Fall keine Abkürzung bzw Referenz erfolgen darf, sondern jedenfalls der volle Name anzuführen ist.

Auf der ersten Seite der Web-Site befindet sich folgender Disclaimer: „<Domainname/>“ erkennt an, dass es sich bei „<Mark/>“ um eine registrierte Marke von <Complainant case=„3“ forceName=„true“/> handelt. Dieser Name wird lediglich zur Identifizierung verwendet. <Domainname/> steht in keiner Beziehung zu <Complainant case=„3“ forceName=„true“/>.“

Die Klägerin ist <Complainant case=„1“/>, welche die Markenrechte für „<Mark/>“ besitzt und diesen Namen für Autos sowie zugehörige Ersatzteile verwendet.

5. Geplante Entwicklungen

Derzeit in Entwicklung befindet sich eine grafische Benutzeroberfläche für den Generator zur Verwendung in Internet-Lernplattformen in der Form eines Applets. Dieses erlaubt es Studenten, einen Fall generieren zu lassen, ihn selbstständig zu lösen, und anschließend die eigene Lösung mit der Musterlösung zu ver- wie auch abzugleichen. Hierbei können die Teilnehmer einzelne Teile der Musterlösung anklicken und dieser Teile ihrer eigenen Lösung zuordnen und sie bezüglich Vollständigkeit und Korrektheit bewerten. Der Grundgedanke hierbei ist, dass eine automatische Korrektur durch den Computer zumindest derzeit noch außerhalb der Möglichkeiten liegt. Um dennoch vollständig autonomes Üben zu ermöglichen, ist diese Selbst-Korrektur vorgesehen, sodass die Rückmeldung detaillierter ausfällt und auch eine virtuelle Note generiert werden könnte.

Erfolgt von einem Teilnehmer eine derartige Zuordnung und Bewertung, so ist es möglich, anschließend adaptiv Fälle mit den Problemkategorien zu generieren, bei welchen Schwierigkeiten auftraten. Hiermit können speziell schlecht beherrschte Gebiete geübt werden, indem bei der zufälligen Generierung des nächsten Falles eine Einschränkung erfolgt, dass zumindest ein Textbaustein aus derselben Klasse (oder einer Subklasse) stammt, welche schlecht gelöst wurde.

Im Rahmen des mit Jahresbeginn gestarteten FWF-Projektes ist es weiters geplant, die adaptive Funktionalität des Generators in Richtung Kooperation mit anderen Lernenden zu erweitern, u. a. durch das Vorschlagen

der Kontaktierung anderer Lernender bei Problemen mit bestimmten Fällen, was ebenfalls auf dem Vergleich der eigenen mit der Musterlösung basiert. Um ein persönliches Lernprofil zu erstellen, werden sowohl die generierte Angabe als auch die Lösung des Teilnehmers in dessen persönlichem Lernordner archiviert, sodass eine Lernhistorie entsteht und ein Nachschlagen möglich wird. Angabe und Lösung können auch öffentlich gemacht werden, was eine anschließende Diskussion darüber mit anderen Teilnehmern erlaubt. Im Falle der Institutionalisierung eines öffentlichen Zugangs wäre es auch möglich, dass Studenten jeweils die Lösungen anderer Teilnehmer bewerten (ohne Einfluss auf die Note der Lehrveranstaltung, ev. auch ein- oder beidseitig anonym), sodass eine unabhängigere Beurteilung erfolgt und gleichzeitig die Kommunikation zwischen Teilnehmern angeregt wird.

Zur Verbesserung der Lesbarkeit der Angabe bzw. der Musterlösung und der Reduktion des Störfaktors eines nicht notwendigerweise logischen Aufbaus (siehe oben) sollen die „Querverweise“ zwischen den Textfragmenten verbessert werden. Konkret bedeutet dies, dass z. B. der Beschwerdeführer nicht durchgängig als „Beschwerdeführer“ bezeichnet wird (was zu einer gewissen Eintönigkeit und häufigen Wortwiederholungen führt), sondern abwechselnd durch seinen vollen Namen oder „er“ bzw. „sie“. Bei letzterer Version ist jeweils zu beachten, dass die Referenz eindeutig ist, was auf dem grammatikalischen Geschlecht der beiden beteiligten Parteien sowie dem bisher letzten Vorkommen im Text basiert.

Zur Überprüfung der Funktionalität sind ausführliche Tests am Institut geplant. Der didaktische Einsatz, insbesondere im Hinblick auf Adaptivität und Kooperation zwischen Lernenden wird jedoch direkt in einer Lehrveranstaltung erfolgen: Rechtsprobleme von Domainnamen spielen dort eine große Rolle und auch in der Abschlussklausur ist immer ein praktischer Fall zur UDRP enthalten. Als Vorbereitung darauf wird das System teilweise Anwendung finden. Durch einen Vergleich der Nutzung des Trainers mit den Ergebnissen beim entsprechenden Klausurbeispiel kann dann die Effektivität des Systems überprüft werden. Zusätzlich wird noch eine Online-Umfrage stattfinden, um die Akzeptanz und die Einschätzung der Lernenden über die Nützlichkeit zu erheben.

6. Ausblick*

Auch wenn es sich in dem Projekt um ein rechtlich begrenztes Gebiet handelt, ist die automatisierte Erzeugung von Übungsfällen schon heute möglich. Problempunkte für eine Generalisierung des Ansatzes bestehen derzeit hauptsächlich in zwei Aspekten:

– Generierung von Fällen, die nicht nur möglich, sondern auch sinnvoll sind:

Eine Überprüfung auf rechtliche/sachliche Möglichkeit ist schon jetzt möglich. Derzeit nicht realisierbar ist eine analoge Prüfung, ob ein Fall real sein könnte oder Lesern zumindest so erscheint.

– Behandlung komplexerer Rechtsgebiete: Hängt die Entscheidung, ob ein Tatbestandsmerkmal erfüllt wird oder nicht, von komplexen Abwägungen oder dem Kontext ab, so funktioniert der vorgestellte einfache Ansatz nicht. Ein Problem ist etwa, wenn die Kombination zweier „kleiner“ Einzелеlemente dazu führt, dass ein Merkmal vollständig erfüllt wird, d. h. in der Berechnung die bloße Addition eines „Erfüllungswertes“ und der Vergleich mit einem Schwellenwert nicht mehr ausreicht (Synergieeffekte).

* Diese Publikation ist das Ergebnis des vom FWF geförderten Projekts ASCOLLA – Adoptive Support for Collaborative E-Learning (P20260-N15).