Research@MICS
Media Computer Science Passau

Prof. Dr. Michael Granitzer

work from

Dr. Christin Seifert (Habil cand.)

Stefan Zwicklbauer (PhD cand.)
  Jörg Schlötterer (PhD cand.)
  Johannes Jurgovsky (PhD cand.)
  Sebastian Bayerl (PhD cand.)
  Stefan John (PhD cand.)
  Albin Petit (PhD cand.)

Lisa Wagner (MsC cand.)
Alexander Treml (MsC cand.)
Stefan Kunz (MsC cand.)
MICS in a Nutshell

The Data

Fields

- (Text) Data Mining
- Applied Machine Learning
- Information Retrieval/NLP
- Visual Analytics / HCI
- Semantic Web
- Social Networks

Projects

1. EEXCESS - Personalised, privacy preserving federated recommendations for cultural and scientific content (EU FP7)
2. CODE (finished) - Fact Extraction and Enrichment from Scientific Articles (EU FP7)
3. MICO - Media in Context – Cross-Media Recommendations and Semantic Representation (EU FP7)
4. BODA - Big and Open Data for SME’s (Bayern)
5. mirKUL - Interaktive Multimedia Videos (BMBF)
6. Industrial Research Project – Credit Card Fraud Detection
Project 1: EEXCESS
FP 7 IP, 10 Partners, Scientific Coordinator, ongoing
http://eexcess.eu/
EEXCESS - Goal

„Contextualised, privacy-preserving access to scientific and cultural long-tail content “

Inject cultural and scientific content into existing web channels

- Websites (Wikipedia, etc.)
- CMS/LMS
- Social media channels (Twitter, etc.)
- Content Consumption and Content Creation Processes

![Graph showing Avg. Monthly Visitors (USA, 2014) vs. Rank of the Web site]
EEXCESS Solution

Involved in
Content Consumption (e.g. Browsing, SNA)

Involved in
Content Creation (e.g. Writing Blogs, Editors)

Recommendation

context
content

content

ZBW Content
Mendeley Content
AMBL Content
Europeana
CT Content
Open Access
EEXCESS – Our Solution

Install: go to Chrome Webstore -> Search for EEXCESS
EEXCESS – Our Research Questions

Paragraph Decomposition and Detection (heuristic)

Privacy Preservation:
• Resource Efficient Text Mining on the Client
• Privacy Preserving Querying (joint PhD Student with INSA Lyon)

Paragraph Summarization + Query Generation

Visual Query Navigation and Search Result Visualisation (+ tons of UI stuff)
Project 1: EEXCESS

Paragraph Summarization and Query Construction

Includes: Prediction Queries using CRF’s; New Test Data Sets; A new state-of-the-art Entity Disambiguation approach
EEXCESS: Paragraph Summarization and Query Generation

Research Question
Can we predict manual queries from a given paragraph?

Experiment 1

- **Given a text selection**, train a linear chain CRF to annotate a word in the selection as query term/not query term.
- Evaluate on collected ground truth data
  - Browser-plugin on Wikipedia
  - Selection + manual queries + ratings + tasks
  - 2499 text selection query pairs

Table 2: Data set overview. Time in minutes.

<table>
<thead>
<tr>
<th>users</th>
<th>tasks</th>
<th>anno</th>
<th>queries</th>
<th>views</th>
<th>ratings</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>217</td>
<td>1332</td>
<td>4562</td>
<td>3267</td>
<td>15043</td>
<td>10252</td>
</tr>
</tbody>
</table>
EEXCESS: Paragraph Summarization and Query Generation

Research Question
Can we predict manual queries from a given paragraph?

Experiment 1 – Results
In Accuracy (%) + Baseline (unbalanced dataset)

Splits:
- 10-folded
- Transfer model across
  - Users
  - Tasks

Conclusion
- Model stable across tasks, not users
- Improvement over (useless) baseline
- Easy task comparable to noun phrase detection in sentences

<table>
<thead>
<tr>
<th></th>
<th>feature set</th>
<th>trivial</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$i, c, t$</td>
<td>$i, t$</td>
</tr>
<tr>
<td>users mean</td>
<td>76</td>
<td>77</td>
</tr>
<tr>
<td>SD</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>tasks mean</td>
<td>82</td>
<td>83</td>
</tr>
<tr>
<td>SD</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>10-fold mean</td>
<td>89</td>
<td>88</td>
</tr>
<tr>
<td>SD</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

$i$ - the identity of a term, i.e. the term itself
$c$ - whether the term begins with upper- or lowercase
$t$ - POS tag

EEXCESS: Paragraph Summarization and Query Generation

Research Question
Can we predict manual queries from a given paragraph?

Experiment 2 – Extend to Paragraphs (currently ongoing)
- Extend query to paragraph
- Challenges
  - Summarize paragraph through semantic features
  - One paragraph contains multiple, potentially correct search queries depending on the individual users preference
  - Models vary across users
  - Gather manual test data according to Marchionini & White’s Information Seeking Model on Wikipedia Pages

Preliminary results of linear Chain CRF
- 90% accuracy with 85% baseline (negative predictor)
- 0.64 precision, 0.44 recall
- Vocabulary has strong influence
EEXCESS: Paragraph Summarization and Query Generation

Research Question
Can we utilize semantic resources (e.g. DBpedia) to improve query generation?

Current Status
• Work mainly done in the field of entity linking/entity disambiguation
• Evaluation in terms of query generation still pending

Now: Focus on (collective) Entity Disambiguation using Word2Vec based Semantic Embeddings

1. What is Word2Vec?
2. How to use Word2Vec for Entity Disambiguation?
3. Does it help? Yes 😊
   • Increase robustness of entity disambiguation approaches
   • New state of the art approach on most data sets
   • Can be made KnowledgeBase Agnostic, i.e. a general preprocessing step for text AND semantic resources (i.e. ontologies/thesauri)
Word2Vec: Neural Network based Language Model in a Nutshell

**Goal:** Estimate $P(w_t|w_{t-1})$ using a d-dimensional vector $v$ per word $w$ (aka Semantic Embeddings) using the softmax function:

$$p(w_0|w_I) = \frac{\exp(v_w \top v_{w_I})}{\sum_{w=1}^{W} \exp(v_w \top v_{w_I})}$$

**Training:**

- Predict the context of a word in a sliding word window
- Optimize Parameters using Stochastic Gradient Descent

$$p(W|w_t) = \frac{\sigma(V \cdot H)}{\sum_{j=1}^{M} \sigma(V_j \cdot H)}$$

$V$ as embedding matrix $|w| \times d$

$H$ as hidden layer $d \times |w|$

$\sigma$ as non-linear activation function (i.e. sigmoid)

$V$ as vector of all Words

- Noise Contrastive Estimation for Speeding up Softmax

$$\log \sigma(v'_w \top v_{w_I}) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} \left[ \log \sigma(-v'_w \top v_{w_I}) \right]$$
**Results:**

**Country and Capital Vectors Projected by PCA**

- China
- Japan
- France
- Russia
- Turkey
- Germany
- Italy
- Spain
- Greece
- Turkey
- Poland
- Portugal
- Austria
- Berlin
- Warsaw
- Athens
- Rome
- Madrid
- Lisbon
- Ankara
- Moscow
- Tokyo

**China - Beijing + Germany = near(Berlin)**

~72% on the Word Analogy Reasoning Task
EEXCESS: Paragraph Summarization and Query Generation

Research Question
How can we accurately map surface forms to entities in a knowledge base?

Approach
1. Preprocessing: Create semantic embedding per entity
   - Word2Vec: word-level embedding
   - Doc2Vec: paragraph level embedding
2. Generate candidate entities \( E \) through index lookup
3. Create a directed, weighted entity graph from all candidates
   i. The weight is based on the mean of the
      • word-level embedding similarity, i.e. \( \text{word2vec}(e_1, e_2) \)
      • Paragraph-level embedding of entity with the surrounding context in the text of its surface form, i.e. \( \text{doc2vec}(e_1, m_1) \)
   ii. The topic node \( e^0_1 \) summarizes already disambiguated entities
4. Solve page rank and take highest ranked entity per surface form (or abstain)
EEXCESS: Paragraph Summarization and Query Generation

Research Question
How can we accurately map surface forms to entities in a knowledge base?

Results (F1 Measure)

<table>
<thead>
<tr>
<th>Data Set</th>
<th>DoSeR</th>
<th>DoSeR (-Doc2Vec)</th>
<th>PriorProb</th>
<th>Wikifier</th>
<th>Spotlight</th>
<th>AIDA</th>
<th>Babelfy</th>
<th>WAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE2004</td>
<td>0.907</td>
<td>0.872</td>
<td>0.831</td>
<td>0.834</td>
<td>0.713</td>
<td>0.815</td>
<td>0.561</td>
<td>0.800</td>
</tr>
<tr>
<td>AIDA/CONLL-TestB</td>
<td>0.784</td>
<td>0.754</td>
<td>0.661</td>
<td>0.777</td>
<td>0.593</td>
<td>0.774</td>
<td>0.592</td>
<td>0.843</td>
</tr>
<tr>
<td>AQUAINT</td>
<td>0.842</td>
<td>0.842</td>
<td>0.820</td>
<td><strong>0.862</strong></td>
<td>0.713</td>
<td>0.532</td>
<td>0.652</td>
<td>0.768</td>
</tr>
<tr>
<td>DBpedia Spotlight</td>
<td><strong>0.810</strong></td>
<td>0.775</td>
<td>0.745</td>
<td>0.797</td>
<td>0.789</td>
<td>0.508</td>
<td>0.522</td>
<td>0.652</td>
</tr>
<tr>
<td>MSNBC</td>
<td><strong>0.911</strong></td>
<td>0.876</td>
<td>0.711</td>
<td>0.851</td>
<td>0.511</td>
<td>0.782</td>
<td>0.607</td>
<td>0.777</td>
</tr>
<tr>
<td>N3-Reuters</td>
<td>0.850</td>
<td>0.810</td>
<td>0.700</td>
<td>0.703</td>
<td>0.577</td>
<td>0.596</td>
<td>0.534</td>
<td>0.644</td>
</tr>
<tr>
<td>IITB</td>
<td>0.741</td>
<td>0.738</td>
<td>0.711</td>
<td><strong>0.766</strong></td>
<td>0.447</td>
<td>0.270</td>
<td>0.470</td>
<td>0.611</td>
</tr>
<tr>
<td>Microposts-2014 Test</td>
<td>0.750</td>
<td>0.704</td>
<td>0.630</td>
<td>0.586</td>
<td>0.453</td>
<td>0.453</td>
<td>0.473</td>
<td>0.595</td>
</tr>
<tr>
<td>N3 RSS-500</td>
<td><strong>0.751</strong></td>
<td>0.713</td>
<td>0.678</td>
<td>0.732</td>
<td>0.622</td>
<td>0.716</td>
<td>0.630</td>
<td>0.682</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>0.816</strong></td>
<td>0.787</td>
<td>0.726</td>
<td>0.768</td>
<td>0.602</td>
<td>0.605</td>
<td>0.560</td>
<td>0.708</td>
</tr>
</tbody>
</table>

→ Semantic Embeddings significantly improve the accuracy
Project 1: EEXCESS
Visualising Search Results and Query Navigation Support
Includes: Two new visualisation: FacetScape and Query Crumbs
EEXCESS: Visualising Search Results

Research Question
Can we provide a better overview over facets of a query?

Approach
1. AW Power Voronoi
2. Tag Layout
3. Interactions (Zoom, Remove, Filter)
4. Comparative user eval

Authors
EEXCESS: Support Query Navigation

Research Question
How can we support re-querying and backtracking?

Approach
1. Developed query history model
2. Bread-crumb like mini-visualisation
   • Query similarity (binary, percentage, detailed)
3. Formative user evaluation
   • Understandable without explanation
   • Usable without explanation
   • Uptake (60% voluntarily choose to use the vis)
Project 1: EEXCESS
Resource Efficient Text Mining on Web-Clients
Includes: Analyse the reduction of word embeddings
Excludes: software development
EEXCESS: Resource Efficient Text Mining on Web-Clients

Research Question
Given the success of word-embeddings (i.e. word2vec), can we reduce their size in order to use them in low-memory environments (i.e. Java Script client)

Approach
• Analyse the effect of different pruning strategies on word embeddings

Research Question
Given the success of word-embeddings (i.e. word2vec), can we reduce their size in order to use them in low-memory environments (i.e. Java Script client)

Results

Datasets/Tasks:
- wa, and QVEC for word analogy task
- WordSim353 and MEN for word similarity

Averaged over dimensionalities are 50, 100, 150, 300, 500

Fig. 2: Mean relative loss of embeddings after (a) PCA: percentage of removed dimensions, (b) Pruning: percentage of removed parameters and (c) Bit-Truncation: remaining Bits. Scores on the QVEC datasets are not shown for PCA since they are not comparable across different word vector sizes.

EEXCESS: Resource Efficient Text Mining on Web-Clients

Research Question
Given the success of word-embeddings (i.e. word2vec), can we reduce their size in order to use them in low-memory environments (i.e. Java Script client)

Results for different dimensionalities

![Graph showing relative loss of embeddings on the syntactic word analogy dataset (wa-syn) after PCA (a), Pruning (b) and Bit-Truncation (c).](image)

Fig. 3: Relative loss of embeddings on the syntactic word analogy dataset (wa-syn) after PCA (a), Pruning (b) and Bit-Truncation (c).

EEXCESS: Resource Efficient Text Mining on Web-Clients

Research Question
Given the success of word-embeddings (i.e. word2vec), can we reduce their size in order to use them in low-memory environments (i.e. Java Script client)

Same idea, but different Learning Approach and Task
- Train Recursive Autoencoder and Convolutional Neural Networks for Sentiment Classification
  - Note: Embeddings are trained implicitly
  - Logistic Regression Layer as Decision Layer
- Prune the embeddings by removing weights with low absolute value
- Observe changes in accuracy
  - Accuracy drop after
    - 90% for CNN
    - 80% for RAE

Fig. 1. Binary sentiment polarity classification accuracy of Logistic Regression. Underlying sentence representations were extracted with pruned versions of CNN or RAE at different pruning levels.

Project 1: EEXCESS

Privacy Preserving Querying

Only a brief overview. Joint work with INSA Lyon (and most parts have been developed there)
Research Question
How can we avoid that a search engine profiles their users?

Approach
Develop a new protocol based realized through an proxy server consisting of two components

1. Unlinkability: Avoid linking the query itself to the user id
   - Encrypt query + key to encrypt results
   - User-id and encrypted query processed by different services

2. Indistinguishability: Hide the true query from the search engine
   - Assume Boolean Queries
   - Generate fake queries
   - OR fake queries with real query
   - Filter OR results based on title

EEXCESS: Privacy Preserving Querying

Research Question
How can we avoid that a search engine profiles their users?

Results
- 2 Attacks based resp. on machine learning and term similarity
- 3 concurrent methods of privacy-preserving Web search (GooPIR, TrackMeNot, TOR)
- Dataset: 343,548 queries from 300 active users - AOL search logs
- Metrics: percentage of queries de-anonymised vs. accuracy of results
  - Accuracy measured in # of rank differences between original and reconstructed results

![Graph showing cumulative frequency and percentage of queries de-anonymized]
EEXCESS – Follow Ups

Zero-effort Querying for Digital Libraries

- In-depth writing style analysis of paragraphs
- Learning boolean queries

Representational Learning

- Learning feature representations instead of feature engineering
- Study semantic embeddings on different kinds of data
  - Networks
  - Time Series Data

Deep Learning for Media Analysis

- How powerful are deep learning methods?
- Efficiency?
- Applicability with small number of datasets?
Project 2: CODE
FP 7 IP, 4Partners, Scientific Coordinator, finished
CODE - Goal

Improve access to facts from scientific literature

- Extract facts from PDFs
- Link those facts to the Linked Open Data Cloud
- Provide decentralised, usable search interfaces
- Provide visual analysis tools for linked data
- Crowd-based quality control

Why?

- “Improvements that don’t add up”
  Armstrong et. al. 2009
- “Why most research results are false”
  Ioannidis, 2005
Extracting Facts from Research Publications

Clustering 

Fact Extraction 

Search Linked Data

Search 

Disambiguation 

Type Inference 

Our Work in the Project
Column Type Inference

Given:

- Table Column: header as type + data in cells \((l_1 \ldots l_i)\)
- Knowledge Base: Set of Entities (E) + Typehierarchie(T)

Goal: Link Columns to Entity

Approach:

![Diagram](image)

**Fig. 1.** Annotation process. 1) Cell labels are disambiguated to entity candidates. 2) Types of entity candidates are determined. 3) Type information is aggregated to determine the header type candidates.
Column Type Inference

Results:

- DBPedia as Knowledge Base (rdf:type, dc:terms)
- 50 Tables from Wikipedia (according to prior work of Limaye et al.)
  - 132 columns, 10-232 rows.
  - Manual annotation of columns (2.5 annotations per column header)
    - 169 rdf:type
    - 169 dc:terms

Table 1. Performance for different cell annotation methods and type vocabularies. Reporting macro-averaged precision $\pi$, recall $\rho$, $F_1$.

<table>
<thead>
<tr>
<th>Vocabulary</th>
<th>$\pi^M$</th>
<th>$\rho^M$</th>
<th>$F_1^M$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rdf-Type</td>
<td>0.24</td>
<td>0.22</td>
<td>0.23</td>
</tr>
<tr>
<td>DublinCore</td>
<td>0.59</td>
<td>0.51</td>
<td>0.55</td>
</tr>
<tr>
<td>Rdf-Type + DublinCore</td>
<td>0.64</td>
<td>0.27</td>
<td>0.38</td>
</tr>
</tbody>
</table>

Discovering and Merging RDF Data Cubes

RDF Data Cube

- Semantic Web Format for representing statistical data (i.e. OLAP Cubes)
- Dimensions, Measures and Attributes

**Given:** RDF Data Cubes residing in decentralised repositories

**Goal:** Discover similar data cubes that can be merged

- E.g. Cube1: Census Uni Passau + Cube 2: Census Uni Weimar

---

Discovering and Merging RDF Data Cubes

Approach

1. Select cube \( c_1 \)

2. Select measure

3. Select matrix aggregation

4. Compute ranking

A: set of available cubes

B: \( c_1 \)

C: top \( k \) ranked cubes

2. Measures to compare cube components

- TFIDF, Word2Vec, Label Similarity, Graph Distance, Concept Similarity

3. Estimate Mergability Value

- Average, Maximum

- Bi-partite Graph Matching for finding optimal pairs

Preliminary Results

- 3 Cubes out of 69 manually annotated cubes

- Best solution: Word2Vec + Concept Similarity with Bi-partite Graph Matching

- Large Variance

<table>
<thead>
<tr>
<th>C1</th>
<th>D1</th>
<th>D3</th>
<th>D4</th>
<th>M2</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>1</td>
<td>0.3</td>
<td>0.1</td>
<td>0</td>
</tr>
<tr>
<td>D2</td>
<td>0.5</td>
<td>0.2</td>
<td>0.9</td>
<td>0.1</td>
</tr>
<tr>
<td>M1</td>
<td>0.1</td>
<td>0.3</td>
<td>0.1</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Mergability Value: 0.9
Linked Data Query Wizard

Problem: Data Quality in the LOD
Web-based Visualisation
CODE - Reflection

- **Extraction Quality:** Some additional steps in extracting facts, but lacking quality yields to low usability

- **Data Quality:** Small parts of the LOD are OK, the rest has extremely low quality. I doubt future uptake.

- **Search for Research Facts:** Still interesting topic, but low uptake due to low LOD quality

- **Crowd based Quality control:** Missing motivation in our scenario. Could not identify a concrete use-case + missing ambitions from our partner to go beyond standard business
CODE – Possible Follow Ups

Fact Extraction from Scientific Literature

- Still unsolved. Needs stronger recognition methods (e.g. Deep Nets)
- Impact dimishing over the next years
  - Newer publications > older publications
  - Research data (+ more semantics) to be published with articles in future

Integrating (Linked) Open Data into Data Mining Processes

- E.g. Weather data in predicting customer behavior
- More focus on machine learning algorithms
  - Augment the data set
  - Optimize models:
    - Specialised Kernels
    - Topology selection in Neural Networks
Other Projects Overview
Credit Card Fraud Detection

Feature Engineering
- Primary Attributes: Date, Time, Amount, Currency….
- Derived Features to aggregate user behaviour: Avg. Amount, Transactions/Day …

Machine Learning Algorithm
- RF>NNet, SVM, LR
- Data set selection as crucial property
- Non-stationary models, unbalanced data sets, incremental learning

Questions
- Can we integrate new features from Linked (Open) Data sources automatically to improve CCRD performance?
- How does Deep Learning compare in terms of accuracy to current ML Algorithms?
- Can Deep Neural Networks work in near real time settings for training and testing?
- How can users/engineers understand decisions taken and their impact?
Analyse Navigation Behaviour in Information Networks

Research Goal

- Agents navigate in an Information Network (i.e. Wikipedia) based on hierarchical background knowledge (in terms of nodes in the network)
- Find the optimal hierarchy of nodes using genetic algorithms

Approach

- Develop recombination mechanisms for trees and according fitness functions
- Study the best hierarchies

Fig. 7. Three Wikipedia hierarchies of different quality. Subfigure (a) shows the best hierarchy created by our experiments, with a global stretch of 8.54. In subfigure (b) the global stretch is slightly lower (11.97). A subtree splitting off from the navigational core can be observed. The last hierarchy has a global stretch value of 22.35. A very linear structure is already clearly noticeable.
Hypervideos for Knowledge Transfer

Knowledge transfer for manual tasks using hypervideos and 2\textsuperscript{nd} Screens (e.g. mobile phones)
Thanks for your attention. Questions?