<Bachelorarbeit> am Institut für Informatik der Freien Universität Berlin

Human-Centered Computing (HCC), AG NBI

Recommender System for Idea-Clustering based on Semantic Similarity of Concepts in Knowledge Graphs

– Exposé –

Luka Stärk

Matrikelnummer: 374532

luka.staerk@campus.tu-berlin.de

Betreuerin: Prof. Dr. C. Müller-Birn

Die Betreuung macht doch Michael. Bitte entsprechend vermerken.

Berlin, October 24, 2019

1 Motivation

The research project Ideas2Market explores the innovation process for applications of new technologies. A central task is to generate many ideas, to cover most possible solutions on how to apply the technology. This procedure is implemented using collaborative innovation approaches to crowd-source ideas. These ideas are not yet fully evolved and considered to be on a brainstorming level, in the following they will be referred to as spark ideas. Nevertheless, these spark ideas introduce great variety and creative value because they are created by different persons with diverse backgrounds. Still, finding valuable spark ideas has proven challenging and due to their large number, it becomes unfeasible to check every spark idea manually and to derive benefits from them for advanced ideas. These ideas are evolved by experts in the further process and then become refined and transformed into product opportunities to deploy onto the market as the last step. The project Ideas2Market aims to solve these problems with software support and by researching the human needs in creative processes. The software supported collaborative-ideation process can be described in three phases as illustrated in Figure 1:

idea sparks

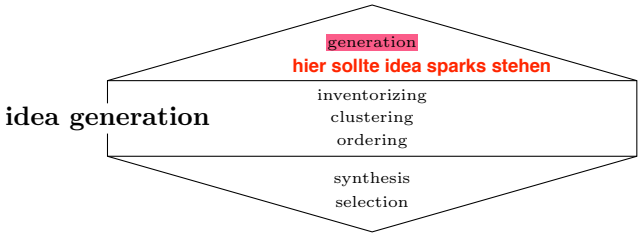

Figure 1: Three Phase Diamond [Tassoul and Buijs, 2007]

1. Divergent Phase — brainstorming and generating spark ideas.
2. Clustering Phase — organizing and making sense of the spark ideas by grouping them on a two-dimensional idea-map.
3. Convergent Phase — synthesize i.e. summarizing and evolving spark ideas to a few novel and promising ideas.

Ich würde die Grafik großer machen und dann diese drei Punkte inkludieren. Ansonsten ist die Grafik unterverständlich.

categories of what?

When clustering, the categories are not always clear to us, the decision of creating a cluster is based on feeling and intuition and can be reversed any time. During the process an ordering emerges and the relationships between spark ideas become more visible, so fare the theory. This process is beneficial as an activity in acquiring a more profound understanding of the idea-space [Siangliulue et al., 2016] and producing more valuable ideas in the Convergent Phase. For growing numbers of spark ideas, it becomes more challenging to organize the idea-space and to take into account all potential spark ideas for one cluster. This task can then be monotonous and time-consuming. My thesis is about counteracting this problem and increasing efficiency in the clustering process with a recommender system that proposes spark ideas based on the selection of content, that can either be a spark idea or single concepts.

process of what?

Die Zielsetzung ist zu diesem Zeitpunkt des Lesens noch ohne weiteres Wissen, unverständlich. Es wurde noch nicht erklärt, warum das Empfehlungssystem gebraucht wird und was ein Konzept sein soll.

Figure 2: Clustering view of the Orchard interface, where the recommender frame (1) displays similar spark ideas to *SPARK 2* (4) from the Cluster named *PET* (3) with two spark ideas. (5) The right column displays the full content of the spark idea, that the user selects by mouse click and further the caption is set in bold, see *SPARK 2* (4).

2 Thematic Classification of Thesis

2.1 Orchard Clustering Application

In the research project Ideas2market the Clustering Web-Application Orchard has been developed to support the second and third phase of the ideation process (see Figure 2 for the clustering step). Orchard is inspired by the IdeaHound project [Siangliulue et al., 2016] and a tool for creative ideation to effectively synthesize ideas from numerous spark ideas. For the clustering phase, the user can drag and drop ideas from the *Spark Stack* on to the whiteboard. To create clusters or add to an existing cluster, the user drops one spark idea on to another or an existing cluster. The user can inspect a spark idea in detail by clicking on it. In that case the complete description and labels of the spark idea are displayed in the right column (see Figure 2 (5)).

2.2 Supporting Effective Collective Ideation at Scale

One solution to increase efficiency in synthesizing ideas is to introduce a predefined idea-map, where the spark ideas are organized in clusters by similarity score [Siangliulue, 2017, 124]. In addition, it is easier for the user to internalize the idea-space and thus interact more with rare ideas [Siangliulue, 2017], which is a benefit in quality because ideas are often mundane or repetitive [Siangliulue et al., 2016]. Then again, the user is more fixated on the categories that were given by the clusters and might miss other possible

syntheses that would have been created without the suggested clusters [Siangliulue, 2017]. To increase efficiency in the idea synthesis and prevent fixation on given categories I propose an idea recommender system.

Ok. Aber es wird trotzdem noch nicht klar, wo genau im Clustering-Prozess der Recommender eingesetzt werden soll.

2.3 Recommender System

Recommender systems (RS) became popular in commercial platforms like Online Shops and Streaming Services and is a subclass of information filtering system, that predicts the user's interest for specific content. A distinction is made between mainly two types. Collaborative filtering where predictions are derived from the behavior of other users and content-based filtering, which is based on the information of the item to recommend and the user's interaction history with the RS. In the case of Orchard, the second applies, the recommendation depends only on the user's current selection of content and information extracted from the spark ideas.

Würden hier nicht am Besten die knowledge-based Ansätze passen?!

In the scope of my bachelor thesis, an RS is developed for the clustering application Orchard to recommend spark ideas that are similar to a selected spark. The similarity measures of the RS are based on a single spark idea or one concept of one spark. In this manner, one can specify further the recommendations made for other spark ideas.

Wenn ich das so lese, wird mir nicht recht klar, warum wir hier von einem Recommender sprechen. Du bestimmst die Ähnlichkeit. That's it.

2.4 Knowledge Graph

For the similarity measures, I use the relations between defined concepts in a Knowledge Graph (KG). In the context of semantic web and linked data many Knowledge Graphs like DBpedia and Wikidata, are freely accessible and gain increasing popularity. KGs are semantic networks where relations between concepts and entities are recorded as triples (subject, predicate, object). These Information Networks are used for different Natural Language Processing and Information Retrieval tasks like word sense disambiguation, topic modeling and Question Answering [Nastase, 2008]. Semantic similarity is a metric to measure the similarity of concepts in the KG through their hierarchical relations [Zhu and Iglesias, 2017]. The advantage of this approach is that the similarity becomes interpretable when looking up the connecting path or the least common subsumer (LCS), which is the lowest common ancestor concept in the directed acyclic graph. In statistical approaches, this information is more difficult to extract because the semantic relationships of concepts are not accessible as facts, like in a KG, rather as distances in high-dimensional spaces. Wikidata records more than 400 million statements and 40 million entities and therefore covers most of the real-world entities and is considered useful as KG in this approach.

2.5 Similarity of Concepts

The publication *Computing Semantic Similarity of Concepts in Knowledge Graphs* of Zhu and Iglesias [2017] discusses different metrics of concept similarity in Knowledge Graphs and compares them to their approach *wpath* with gold standard data sets of human-judged similarity in meaning. Their metric *wpath* for measures of similarity

wird nicht erklärt! Beispiel?

considers shortest-path length between two concepts in the Knowledge Graph and the Information Content (IC) of their least common subsumer (LCS). The IC of a concept determines how abstract or specific a concept is and how much information the entities of a concept share, so more abstract concepts hold lower IC values and more specific once higher values of IC. There are different ways of measuring the IC, Zhu and Iglesias [2017] propose two of them:

woher kommt der negative log? Welcher log (Basis) ist es?

Definition 2.1. Information Content corpus-based:

Let c_i be a concept, then the $IC_{corpus}(c_i) = -logProb(c_i)$. Given a large general text-corpus $Prop(c_i)$ is the probability to encounter a word from the set of $words(c_i)$ that are subsumed or associated with c_i. $Prob(c_i) = \frac{\Sigma_{w \in words(c_i)} count(w)}{N}$, where N is the total number of concepts observed in the text-corpus.

w?

Definition 2.2. Information Content graph-based:

Let c_i be a concept, then the $IC_{graph}(c_i) = -logProb(c_i)$, where the $Prob(c_i) = \frac{count(entities(c_i))}{N}$ and $entities(c_i)$ is a set of entities with type c_i in the KG and N is the total number of entities in the KG.

Das halte ich in unserem Fall nicht für sinnvoll. Ok., wenn ich eine irre kleine Zahl habe und dann log(10) mache, dann ist es sinnvoll. :)

The parameter to weight between IC of the LCS and the path length between concepts is $k \in (0, 1]$.

$$sim_{wpath(c_i,c_j)} = \frac{1}{1 + length(c_i, c_j) \cdot k^{IC(lcs)}}$$

??? Was soll das jetzt? Was ist lcs? Warum wird das jetzt potenziert ..

3 Goal

warum ist diese Ansatz insbesondere geeignet?

The goal of my thesis is to improve the clustering process in the Orchard application with a recommender system. Therefore, concept similarity is calculated for all concepts extracted from 60 spark ideas. The measure of similarity for concepts is Knowledge Graph-based and calculated through the metric *wpath* of Zhu and Iglesias [2017]. Given the measures of similarity between concepts, the similarities between concepts and ideas, and the Word Mover's Distance between ideas are calculated, these are the measurements needed to integrate the recommender system into the Orchard clustering application.

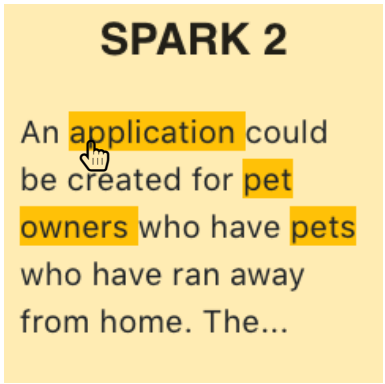

Figure 3: Spark with highlighted concepts

In Orchard, as Figure 3 illustrates, the user can click on a spark idea or a highlighted concept of a spark idea's description to select the content the recommender system calculates similar spark ideas for. The recommended spark ideas are displayed in the recommender frame sorted by highest similarity and the user can scroll through and drag them onto the whiteboard (see Figure 2 (1)).

4 Planned Procedure

The Procedure for my thesis is divided into three substantial parts: implementation, integration into the Orchard application and the validation of the results.

4.1 Implementation

The requirement for the input data of spark ideas is that the concepts of the idea descriptions are assigned to Wikidata items. In prepossessing all stop-word concepts and concepts that do not connect to get KG are excluded. ?? das ist doch hier nicht notwendig.

The following functions will be implemented to apply the *wpath* metric for concept similarity.

- A function to generate a subgraph of the Wikidata Knowledge Graph. The subgraph is extracted with *SPARQL* queries to the *Wikidata* SPARQL-endpoint. The set of concepts that occur in the spark ideas build the bottom child layer of the KG. All existing connections in Wikidata from each idea-concept to *entity*, over the three predicates *subclass of (P279), instance of (P31), part of (P361)* get extracted for the KG. In which *entity* is the highest concept in the hierarchy of the Wikidata KG.
- A function to resolve an directed graph into an directed acyclic graph (DAG).
- A function to calculate the Information Content graph-based for all concepts in the DAG, over the edges *subclass of (P279), instance of (P31).*
- A function that finds the least common subsumer for any two concepts in the DAG.
- A function that calculates the all-shortest ancestral distance in the DAG, i.e. the shortest path between two nodes over their LCS.

For the similarity measures from a concept to a spark idea, I will implement a function that returns the spark ideas which contain the concepts that are most similar to the concept given as input.

To calculate the similarity between spark ideas I will implement the word mover's distance (WMD) [Kusner et al.] based on the similarity of their concepts.

4.2 Integration into Orchard

For the Orchard application, I will integrate the similarities, provided by the implementation for a given data-set of spark ideas, into the database of Orchard. For the client-side I will add a recommender frame (see Figure 2 (1)) and highlight the annotated concepts in the idea-description in the detail view (see Figure 2 (5)). When the user hovers over any spark idea on the whiteboard the concepts in the description become highlighted as well (see Figure 3). The highlighted concepts are clickable and a function updates the recommendations for the new source.

4.3 Validation

The validation consists of the following three steps:

1. To validate the measures of concept similarity, the implementation of *wpath* will be compared to the *R&G* dataset [Rubenstein and Goodenough, 1965] of human-judged word similarity and the results of the implementation of *wpath* from Zhu and Iglesias [2017]. *R&G* is a widely used dataset containing human assessment of word similarity for 65 word-pairs of common English nouns. Zhu and Iglesias [2017] uses the same dataset for *wpath* with corpus-based IC and graph-based IC with the DBpedia ontology. This comparison will be done for different parameter $k \in (0, 1]$ for *wpath*, to optimize the measurements for *R&G*.

2. To find the best weight k between path length and IC for the Wikidata KG, the *wpath* similarity will be applied on several idea-concepts for different parameters $k \in (0, 1]$.

3. As the last step, the recommender system will be tested with Orchard on a qualitative level with three or more persons who are familiar with the clustering process, to evaluate the research question, if the recommender system in Orchard is a helpful feature to effectively find clusters and creating a satisfying result.

5 Technical Implementation

The software is realized in python and the Knowledge Graph is extracted from Wikidata. For the calculation of the corpus-based IC, the implementation of Zhu and Iglesias [2017] is used. The Interface of the recommender system in Orchard is implemented in JavaScript, React and Redux.

6 First Schedule

28. October	Implementation of IC_{graph}
6-10. November	Presentation of the thesis topic at HCC
8. November	Find a primary supervisor from TU-Berlin.
8. November	Annotate and connect $R\&G$ concepts to Wikidata
15. November	Registration of the bachelor thesis at the TU-Berlin
15. November	Implementation of $wpath$
20. November	Validation of $wpath$ and comparison with dataset of Rubenstein and Goodenough [1965] and implementation of Zhu and Iglesias [2017].
30. November	Implementation of WMD for similarity measures.
4. December	Test concept similarity implementation of $wpath$ for Wikidata, to select best parameter k.
20. December	Finish technical implantation, take measurements for similarity.
30. December	The recommender system is integrated into Orchard for gold standard ideas.
3. January	Discussion of the results with Prof. Dr. Claudia Müller-Birn, Michael Tebbe and Maximilian Mackeprang.
25. January	The thesis are written, time for correction.
15. February	Submission of the bachelor thesis

References

Matt J Kusner, Yu Sun, Nicholas I Kolkin, and Kilian Q Weinberger. From Word Embeddings To Document Distances. page 10.

Vivi Nastase. Topic-driven Multi-document Summarization with Encyclopedic Knowledge and Spreading Activation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 763–772, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=1613715.1613812`. event-place: Honolulu, Hawaii.

Herbert Rubenstein and John B. Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, October 1965. ISSN 00010782. doi: 10.1145/365628.365657. URL `http://portal.acm.org/citation.cfm?doid=365628.365657`.

Kanya (Pao) Siangliulue. Supporting Effective Collective Ideation at Scale. May 2017. ISSN http://nrs.harvard.edu/urn-3:HUL.InstRepos:40046559. URL `https://dash.harvard.edu/handle/1/40046559`.

Pao Siangliulue, Joel Chan, Steven P. Dow, and Krzysztof Z. Gajos. IdeaHound: Improving Large-scale Collaborative Ideation with Crowd-Powered Real-time Semantic Modeling. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16*, pages 609–624, Tokyo, Japan, 2016. ACM Press. ISBN 978-1-4503-4189-9. doi: 10.1145/2984511.2984578. URL `http://dl.acm.org/citation.cfm?doid=2984511.2984578`.

Marc Tassoul and Jan Buijs. Clustering: An Essential Step from Diverging to Converging. *Creativity and Innovation Management*, 16(1):16–26, 2007. ISSN 1467-8691. doi: 10.1111/j.1467-8691.2007.00413.x. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8691.2007.00413.x`.

G. Zhu and C. A. Iglesias. Computing Semantic Similarity of Concepts in Knowledge Graphs. *IEEE Transactions on Knowledge and Data Engineering*, 29(1):72–85, January 2017. ISSN 1041-4347. doi: 10.1109/TKDE.2016.2610428.

Anhang I: Auszug Prüfungsordnung Bachelor

FU-Mitteilungen

§ 5
Bachelorarbeit und mündliche Abschlussprüfung

(1) Die Bachelorarbeit soll zeigen, dass die Studentin oder der Student in der Lage ist, ein Thema aus dem Bereich der Informatik unter Anleitung nach wissenschaftlichen Methoden in einer vorgegebenen Zeit zu bearbeiten und seine Arbeit und die Ergebnisse selbständig darzustellen, wissenschaftlich einzuordnen und zu dokumentieren.

(2) Die Bearbeitungsdauer einer Bachelorarbeit beträgt zwölf Wochen.

(3) Studierende werden auf Antrag zur Bachelorarbeit zugelassen, wenn sie

1. die Module
   - Datenstrukturen und Datenabstraktion
   - Grundlagen der Theoretischen Informatik
   - Logik und Diskrete Mathematik
   - Analysis oder Analysis I
   - Lineare Algebra oder Lineare Algebra I sowie
   - Rechnerarchitektur

   erfolgreich absolviert haben,
2. im Bachelorstudiengang Informatik zuletzt an der Freien Universität Berlin immatrikuliert gewesen sind.

(4) Dem Antrag auf Zulassung zur Bachelorarbeit sind Nachweise über das Vorliegen der Voraussetzungen gemäß Abs. 3 beizufügen, ferner die Bescheinigung einer prüfungsberechtigten Lehrkraft über die Bereitschaft zur Übernahme der Betreuung der Bachelorarbeit sowie eine Erklärung, dass die oder der Studierende nicht an einer anderen Hochschule im gleichen Studiengang, im gleichen Fach oder in einem Modul, welches einem der im Bachelorstudiengang Informatik studierten Modulen vergleichbar ist, Leistungsnachweise endgültig nicht erbracht oder Prüfungsleistungen endgültig nicht bestanden hat oder sich in einem schwebenden Prüfungsverfahren befindet. Der zuständige Prüfungsausschuss entscheidet über den Antrag.

(5) Der Prüfungsausschuss gibt in Abstimmung mit der Betreuerin bzw. dem Betreuer das Thema der Bachelorarbeit aus. Thema und Aufgabenstellung müssen so beschaffen sein, dass die Bearbeitung innerhalb der Bearbeitungsfrist abgeschlossen werden kann. Ausgabe und Fristeinhaltung sind aktenkundig zu machen.

(6) Als Beginn der Bearbeitungszeit gilt das Datum der Ausgabe des Themas durch den Prüfungsausschuss. Das Thema kann einmalig innerhalb der ersten drei Wochen zurückgegeben werden und gilt dann als nicht ausgegeben. Ausnahmsweise kann der Prüfungsausschuss auf begründeten Antrag im Einvernehmen mit der Betreuerin bzw. dem Betreuer die Bearbeitungszeit der Bachelorarbeit um bis zu vier Wochen verlängern. Bei der Abgabe hat die bzw. der Studierende schriftlich zu versichern, dass sie bzw. er die Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt hat.

(7) Die Bachelorarbeit ist von zwei Prüfungsberechtigten zu bewerten, die vom Prüfungsausschuss bestellt werden. Einer der beiden Prüfer soll die Betreuerin bzw. der Betreuer der Bachelorarbeit sein. Mindestens einer der beiden Prüfer muss dem Kreis der Professorinnen und Professoren des Instituts für Informatik angehören.

(8) Die Ergebnisse der Bachelorarbeit werden im Rahmen einer mündlichen Abschlussprüfung, bestehend aus einem etwa 15-minütigen Vortrag mit anschließender etwa 15-minütiger Diskussion und Prüfungsgespräch, vorgestellt und wissenschaftlich eingeordnet und verteidigt.

(9) Voraussetzung für die Teilnahme an der mündlichen Abschlussprüfung ist die Abgabe der Bachelorarbeit. Der Prüfungstermin wird rechtzeitig in geeigneter Form bekannt gegeben.

(10) Die mündliche Abschlussprüfung wird von denjenigen Prüfungsberechtigten, welche die Bachelorarbeit bewertet haben, abgenommen.

(11) Ist die Note der Bachelorarbeit oder die Note der mündlichen Abschlussprüfung nicht mindestens „ausreichend" (4,0), so dürfen Bachelorarbeit und mündliche Abschlussprüfung einmal wiederholt werden.

84 FU-Mitteilungen 6/2007 vom 08.02.2007

Figure 4: Auszug Prüfungsordnung Bachelor