The Role of Semantic Similarity for Intelligent Question Routing

Bojan Furlan*, Slavko Žitnik†‡, Boško Nikolić* and Marko Bajec†
*University of Belgrade
School of Electrical Engineering
Bulevar kralja Aleksandra 73
RS-11120 Belgrade
Email: {name.surname}@etf.bg.ac.rs
†University of Ljubljana
Faculty of computer and information science
Tržaška cesta 25
SI-1000 Ljubljana
Email: {name.surname}@fri.uni-lj.si
‡Optilab d.o.o.
Župančičeva 8
SI-5270 Ajdovščina

***Abstract*—Intelligent Question Routing Systems (IQRS) serve as a knowledge exchange medium in an arbitrary field of expertise, where intensive communication between users is required. The benefit coming from deployment of such systems includes: (a) reducing unnecessary "pinging" of experts, which are a valuable resource and (b) increasing the system owners' (e.g. enterprise, government, university) quality of service, since users are more satisfied with answers, because their questions are answered by the right persons. In this paper we investigate the role of semantic similarity for each stage of IQRS process. For question and answer analysis we use semantic enrichment, more precisely semantic query expansion with ConceptNet, WordNet (Antelope), and SemNet, as well as TF-IDF and IQRS system features. Also, for question routing stage we proposed an algorithm used for calculating semantic similarity between question and profile. Finally, for evaluation we used subset of Yahoo! Answers L6 dataset from which we extracted three different types of users: (1) Top questioneers, (2) Top answerers, and (3) Top questioneers and answerers at the same. Based on carried experiments we found that, besides expertise, interest profiling can improve system performances.**

I. Introduction

The key functionality of Intelligent Question Routing Systems (IQRS) is that for a question addressed by a user, to provide a good answer by searching the list of all (other) available users. The answer is provided by selecting a certain number of competent users (experts) and forwarding them the question. Selected users can provide an answer, which is then returned to the user that posed the question. The survey of state of the art in IQRS is given in [1] which introduced a presentation paradigm that generalizes the essence of approaches found in the open literature. The presentation paradigm includes three basic processing stages related to the three major problems of system implementation: question analysis, question forwarding, and users' knowledge profiling. All approaches presented in the survey are analyzed regarding these three basic processing stages. The outcome of the analysis was a proposal for new approaches that tackle identified problems and the work presented in this paper is a consequence of this research.

Questions are usually short in length and even may be ambiguous. Therefore, as a format for acquisition, storing and representation of information related to questions or user profiles, we have chosen an approach in which the detected concepts are presented in the form of concepts cloud (similar to TagCloud visualization) [2], [3]. One advantage of this approach is that "a significant concept has assigned a higher weight," which provides an intuitive idea of specific relationships between concepts and their importance in the question. Therefore, the generated concept cloud represents a set of information that describes the question. Concepts together with their weights in this set make the specific context in a broader sense which represents a fingerprint of processed text. This fingerprint is specific to each question and it is similar for questions with the same subject and the same meaning. Finally, this fingerprint reveals specific relationships between questions, and the relationship between the question and the topics which the question touches. On the other hand, for each user IQRS system is maintaining profile which is represented in the same way, describing user's interests based on his/her questions and answers. Therefore, identified concepts from questions and user profiles are represented as pairs (keyword, weight). This pair is further referred as a *concept*.

The task of finding a competent user is done by comparing the information extracted from the question with all available user profiles, giving a ranked list of users or "candidates to answer". Based on this ranked list one or more users can be selected, which then should be contacted for an answer. Since the information extracted from the question as well as those maintained in the user profiles are presented in the same way, as lists of concepts, comparison of these two lists is reduced to the calculation of their similarity. Determining the similarity can be carried out by exact comparison, i.e. determining exact matching between words, or by calculating the semantic similarity. Since the question can be semantically very similar to the profile, but still lexically very different, the better results can be achieved by using the semantic similarity. As an example that illustrates this point we can use the appearance of synonyms, words with the identical or very similar meaning, but very different in its form (e.g. words intelligent and smart). Therefore, the main focus of the work presented in this paper is calculation of semantic similarity between the question and the user profile, as well as semantic information extraction from questions and answers in order to calculate this similarity.

The rest of the paper is organized as follows. In Section II we give a brief overview of related work and available technologies for calculating the semantic similarity of short texts. Next, we introduce proposed P2Q (Profile-to-Question) algorithm for determining the numerical similarity score between the question and the user profile. In Section IV we explain different concept extraction types and sources that we use. In the next section we evaluate our approach on a Yahoo! Answers dataset, present the results of executions using various settings and in Section VI we give the conclusion.

II. Related Work & Available Technologies

Based on the analysis of existing approaches for semantic similarity of short texts (or short text semantic similarity - STSS) it was concluded that none of the available solutions can be directly applied for determining the similarity between questions and user profiles. Therefore, determining the similarity is done by using a bag-of-words approach based on a modification of LinSTSS approach [4], in which for all concepts in the question Q we find the most similar matching concepts in the profile P. We decided to base our approach on LinSTSS for the following reasons:

1) This approach includes weights assigned to each compared word, i.e. can naturally deal with concepts.
2) It does not rely on any external knowledge base (e.g. WordNet), manually created inference rules or specific linguistic tools, which would be an obstacle in working with languages that lack these resources. Furthermore, the LinSTSS approach does not use the semantic similarity measure alone, but includes string similarity measure as well, so it gives better results for different forms of infrequent proper nouns, which is one of the major shortcomings of the knowledge-based approaches [5].
3) Finally, LinSTSS is inspired by the method proposed in [6] that relies on a similar bag-of-words approach, but uses word specificity as in [7] to weight the word similarity. However it overcomes the problem of method [7], which has a tendency to overestimate text similarity because it allows multiple words from one text to be paired up with a single word in the other text.

The last item is significant in determining the semantic similarity of two short texts, since it is important to determine which pairs of sentences are semantically similar, but also which are different. However, we wanted to explore if this stands also for the task of finding competent users, i.e. calculating similarity between the question and the user profile, because in this case it is necessary to determine just which profile is the most similar to the question, while it is not necessary to determine those that are different. To some extent this relates to one-class-classification problem, since we only have positive examples, e.g. we are aware of the user that provided the best answer to the question, but negative examples are unknown, i.e. we don't know which users are not able to provide the best answer. Therefore, we introduced modification named P2Q to determine the highest (maximum) similarity between concepts from the question to concepts from the profile, but not vice versa.

The rest of the section gives a brief description of algorithms and tools that are used for user profiling, i.e. analyzing questions and answers. To be able to measure similarity between questions and user profiles, we had to extract as many semantic concepts about users and their behaviour, as possible. In this study we used the following techniques and sources:

- **ConceptNet** [8] is a semantic knowledge base that describes general human knowledge. It includes words and common phrases from many written texts. They are related through open domain predicates and through common knowledge. The database was created manually and partially automatically from Wiktionary and ReVerb system, which is an open information extraction tool that extracts binary relationships of type *phrase-relation-phrase* in an unsupervised manner. The whole database contains 414 thousand English concepts and 903 thousand relationships between them.

- **Antelope** (Advanced Natural Language Object-oriented Processing Environement) [9] is a natural language processing framework (NLP) that can handle large corpora and consists of many extensible modular components. It uses an extended lexicon version of WordNet lexicon with improved conceptualization and integrates a higher level formal ontology. The main components offer syntactic and semantic analysis of texts, anaphora extraction, word sense disambiguation and paraphrase extraction. It also includes access to other opensource NLP libraries such as Stanford Parser, WordNet and VerbNet.

- **SemNet** (Semantic Network of Terms) [10] is a large-scale network of technical terminology which allows querying terms and retrieving ranked lists of their semantically related terms. The network was automatically constructed based on the noun terms from English Google Books Ngram Dataset using word co-occurrence analysis. The network consists of 2.8 million distinct single and multi-word terms and 37.5 million LinSTSS edges between them. SemNet includes a large part of the same concepts and relationships from similar semantic knowledge bases such as WordNet [11] and ConceptNet [8].

- **TF-IDF** (term frequency-inverse document frequency) is a general numerical statistic that defines the importance of each word in a document collection. It is often used in information retrieval and text mining as it gives good string-based performance.

III. P2Q Semantic Similarity

This section describes phase-by-phase the algorithm named P2Q, for determining the numerical similarity score between the question and the user profile. The source code is accessible in publicly available repository[1].

1. *Preprocessing* begins with the text cleaning procedure which deletes all text characters not belonging to the native script of the language in question, removes numbers and words that contain numbers, eliminates punctuation marks, removes stopwords, shifts all capital letters into lower case and finally, lemmatizes them. Processing then continues with the removal of stop words from the given texts and the remaining words from both texts are then stemmed. After this stage, if there are concepts where their keyword consists of more than one word, for each word is created new concept with the same weight as the original one, e.g. from concept (botanical gardens, 0.5) two concepts are created (botanical, 0.5) and (gardens, 0.5). If there are multiple concepts with the same keyword they are merged into one by calculating resulting weight with the probabilistic T-conorm:

$$\mathrm{sum(a,b)} \;=\; a + b - a \cdot b \tag{1}$$

After the preprocessing input data, the question is represented with an array Q given in (2) and the profile P is given in (3). Q consists of pairs $(q_i, w(q_i))$ representing concepts, where q_i is the keyword and $w(q_i)$ is its assigned weight. Similarly, P is consisted of pairs $(u_i, w(u_i))$. The length of array Q is m and for P is n.

$$Q = \{(q_1, w(q_1)), (q_2, w(q_2)), \dots (q_m, w(q_m))\} \tag{2}$$

$$U = \{(u_1, w(u_1)), (u_2, w(u_2)), \dots (u_n, w(u_n))\} \tag{3}$$

2. *String similarity matrix construction* creates a matrix M_1 with dimensions $m \times n$ in which every cell is occupied by a numerical value α, which lies between 0 and 1 representing the string similarity between the column-word and the row-word. The rows of the matrix are used for the words from the question, while the columns represent the words from the profile. A zero value denotes entirely different string contents, while a value of one indicates a perfect string match. The approach which we used to calculate string similarity is the same as in [4] by calculating linear combination of 3 normalized longest common subsequence measures.

$$M_1 = \begin{pmatrix} \alpha_{11} & \cdots & \alpha_{1n} \\ \vdots & \ddots & \vdots \\ \alpha_{m1} & \cdots & \alpha_{mn} \end{pmatrix} \tag{4}$$

3. *Semantic similarity matrix construction* creates a matrix M_2 with dimensions $m \times n$ in which every cell is occupied by a numerical value β which lies between 0 and 1 representing the semantic similarity between the column-word and the row-word. The rows of the matrix are used for the words from the question, while the columns represent the words from the profile. Similar to the string similarity measurement, a zero value denotes entirely different semantic contents, while a value of one indicates a perfect semantic match. We gain the semantic similarity of words in a pair by calculating the cosine similarity in the same way as in [4] by applying COALS algorithm.

$$M_2 = \begin{pmatrix} \beta_{11} & \cdots & \beta_{1n} \\ \vdots & \ddots & \vdots \\ \beta_{m1} & \cdots & \beta_{mn} \end{pmatrix} \tag{5}$$

4. *Similarity matrix unification* combines the string and the semantic similarity matrices into one by multiplying their values by a certain ponderation factor and adding them up as in (6). We used ponderation values of 0.45 and 0.55 for the string and semantic similarity scores, respectively.

$$M_3 \;=\; \psi M_1 + \varphi M_2 \tag{6}$$

$$1 \;=\; \psi + \varphi \tag{7}$$

$$M_3 = \begin{pmatrix} \gamma_{11} & \cdots & \gamma_{1n} \\ \vdots & \ddots & \vdots \\ \gamma_{m1} & \cdots & \gamma_{mn} \end{pmatrix} \tag{8}$$

5. The *final similarity score calculation* start with the unified similarity matrix. The goal is to match words across the two sets according to their mutual similarity score. Hence, we search for the highest value within the final similarity matrix, but for P2Q approach this step is executed differently than in LinSTSS, since we wanted to determine the highest (maximum) similarity between concepts from the question to concepts from the profile, but not vice versa. Therefore, within the unified similarity matrix we search for the highest γ_{ij} value, but only within the each row, since rows refer to concepts from the question. This maximal value for the i-th row is $\max_i(\gamma_{ij})$, where $i \in 0,..m$. After this, for each row this value is multiplied with its normalized weight $w(q_i, u_j)$ and summed. Normalized weight $w(q_i, u_j)$ is calculated as follows:

$$w(q_i, u_j) = 2^{w(q_i) \cdot w(u_j) - 1}, w(q_i), w(u_j) \in (0, 1] \tag{9}$$

where $w(q_i)$ is the weight of the concept q_i from the question and $w(u_j)$ is the weight of the concept u_j from the profile. By using this technique of normalization we can obtain normalized values for each pair of concepts in the range of $(0.5, 1]$. A pair of concepts that has assigned a low importance, and therefore weights with the value close to 0, will have a normalized weight near 0.5, while for those with the high weights, normalized weight will be close or equal to 1. In effect, when using normalized concepts weight in a similarity score ponderation, for pairs made up of important concepts will retain their full similarity score (or almost full), while the scores of pairs containing not so important concepts will be reduced by as much as 50%.

Finally, similarity score calculation is performed by utilizing the following formula:

$$S(Q, U) = \frac{1}{m} \sum_{i=0}^{m} w(q_i, u_j) \cdot \max_i(\gamma_{ij}) \tag{10}$$

IV. User Profiling

There are many approaches to extract information from textual documents. The advantages of statistical approaches are mostly higher precision on large corpora, easier implementation and longer history of research. On the other hand,

[1] https://bitbucket.org/bfurlan/semsim

if we need to process a short text, methods from the field of computational linguistics will give the best results. In order to build efficient user profiles, we collect a number of concepts from user's posts. We separate concepts by source, which can origin from question body, answer body, question title, whole post thread or IQRS system. The more detailed description of data set that we used is given in the next section. The source code for data manipulation and profiling is also publicly available[2].

Before the concept extraction we first employ some preprocessing techniques. This step is carried the same way as in step 1 of the proposed algorithm (Section III). From these we further extract concepts of the following types:

Concept extractor (CE): In our implementation of concept extractor we integrated Antelope and ConceptNet (Section II). Since the Antelope tool is based on a much smaller but more precise WordNet lexicon, it is designed to recognize named entities (e.g., names of people, organizations, states and cities) with high precision, but with limited number of concepts. Together with integrated context extraction it can contribute to better identify concepts. On the other hand, ConceptNet contains much richer semantic network, which provides better results in identifying existing and related concepts, but it does not support named entity recognition or context extraction. When we combined these tools (Antelope and ConceptNet) during the query expansion, we got better results than using them individually.

Since both tools, in addition to the identified concepts, determine the weight of the extracted concept within the range $(0,1]$, we use the following approach to combine both values: If only one of the tools finds some concept, it retains the weight, otherwise we calculate the resulting weight using the probabilistic T-conorm (1).

In order to obtain better results and reduce the number of incorrectly identified concepts we also use the following rules:

1) We empirically defined linear relationship between the minimum weight of a concept (i.e., threshold) and a length of the text (#words_in_text):

$$\min(\text{weight}) = 0.875 + \#\text{words\_in\_text} \cdot 0.125$$

If the weight of a concept is less than the minimum weight, it is removed and therefore not used in further calculations.

2) Since it is much harder to detect the topic from the long text rather than from the short text, which consists only of important concepts, the process of context identification is carried as follows: (1) We first detect all the concepts using Antelope and ConceptNet. (2) Then we again use Antelope to find context for all the concepts from the previous step, considering higher precision.

SemNet (SN) allows for searching ranked lists of semantically related terms. For each extracted word in the preprocessing step we select three most semantically related terms as SemNet concepts. For example, the word "car" is within SemNet described with the following concepts: ("front", 0.038), ("side", 0.024) and ("truck", 0.024).

[2]https://github.com/szitnik/InterestMining

TF-IDF (TFIDF) is a standard measure for weighting words within text documents. Intuitively the word has different weight depending on the source of occurence. Therefore we calculate TF-IDF measures separately for question titles, question bodies and answers. For example, a word "car" has the following TF-IDF weights[3]: (1) 0.163 in the question title, (2) 0.066 in the question body, and (3) 0.0246 in the best answer.

Categories: Each post also contains associated categories. In our dataset there are three hierarchical category types. Their values are more thoroughly explained within the dataset description (Section V-A).

V. Evaluation

This section contains an overview of available datasets and the one that we created from Yahoo! Answers L6 dataset, and as well results of evaluation and discussion.

A. Dataset

We reviewed many question answering (QA) systems and datasets. Finally, we decided to use Yahoo! Answers Webscope L6 dataset [12] because it contains a broad range of distinct question categories and there are enough active users in each of the categories that we selected. Other datasets could be extracted from sites such as AskVille (http://askville.amazon.com), Mahalo (http://mahalo.com), Quora (http://quora.com) and StackOverflow (http://stackoverflow.com). The latter also includes a lot of users but it does not have specified specific categories and more importantly, the whole dataset is mostly single domain oriented. Furthermore, there are lots of other systems like AllExperts (http://allexperts.com), Ask.com (http://ask.com), Answers.com (http://answers.com) and others that lack of one or more public features to build useful dataset.

Yahoo! Answers is a QA site where people post questions and answers, which are publicly available to any web user. The L6 dataset was collected from this site in 2007 and it includes all the questions (i.e. 4483032) and their corresponding answers. Next to these data some anonymized metadata is included, so that we can extract concepts related to the each user.

Questions and answers represent instances of post type and a question with all available answers forms a thread. Each post instance consists of text in the body, selection of main category, category and subcategory and the id of the user who wrote the post. Questions additionally contain also a title. For answer posts, a owner user is known only if this post was considered as the best answer for the question, which imposed the limitation that only for the users that provided the best answer we can relate extracted concepts.

Depending of the post type from which concepts are extracted we distinguish Question, Answer, and Thread type of concepts. User instances are completely anonymized, so we modeled them using their id number. Also, for each user we counted how many answers or questions he/she posted in specific category type.

From the full dataset we extracted three database types:

- **Type 1**: This dataset models interest as it contains users that asked at least ten questions and each of these questions must have at least five answers.

[3]The example was taken calculated on the post 1512658 from the L6 dataset.

- **Type 2**: To model knowledge, we extracted users who best answered at least ten questions. Again, each question thread needed to have at least five answers.
- **Type 3**: To jointly represent interest and knowledge, we extracted users that asked at least five questions and best answered at least five questions. Also, each of these threads needed to have at least five answers.

The requirement to address only the questions that have at least five responses provides a certain level of question quality (eg, the question is not trivial and it attracted the attention of other users). Each database type contains 100 users which are distinct across the dataset. As there are many different categories in the original dataset, we representatively selected five of them and extracted 100 users for each dataset. Distribution of users per each category is shown in Table I. Also, in all three database types, for each user we also selected one additional question for which he/she provided an answer that was considered as the best answer for that question. This question is then used for evaluation as explained in the following section.

Table I: Distribution of post categories in each database type

Category	Number of selected users
Society & Culture	35
Food & Drink	35
Computers & Internet	15
Travel	10
Cars & Transportation	5
Total	100

B. Results and Discussion

To evaluate our work we used mean reciprocal rank (MRR) and Precision@N (P@N) which are widely used in the field. They are defined as follows: (1) MRR is a measure to evaluate information retrieval task in which a list of possible responses to a query (question) is ordered by a probability of correctness. The score is defined in (11) as the average of the reciprocal ranks for a set of questions Q, where for each question q from the dataset we determine the similarity $S(q,u)$ to all available user profiles ($u \in U$) and accordingly rank users. Finally for each question we calculate the rank of the user that provided the best answer ($rank_i$).

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}. \tag{11}$$

(2) Precision is the fraction of the retrieved documents (users) that are relevant to a query (question). Here we count as a relevant user the one that provided the answer to the question that was selected as the best one. P@N is therefore precision which is evaluated at a given cut-off rank N (i.e. considering only top N results). In our domain it measures, for a set of questions Q, the proportion of correctly selected users (relevant users) among all users or intuitively a probability of how likely asking among the top N selected users will result in getting a correct answer. Again, users are ranked here by $S(q,u)$.

Results of evaluation for all 3 database types are given in Fig. 1-4 and all MRR and P@N execution results are available online[4]. By I&I-STSS we refer to the approach proposed in [6], which is not using word weighting to calculate semantic similarity. By LinSTSS we refer algorithm proposed in [4] and for P2Q we refer to the algorithm that we propose in Section III. Since database Type 1 contains only questions posted by evaluated users, from this type we extracted only Question concepts, while for Types 2 and 3 we extracted both Answer (which refers to the best Answer) and Thread (which refers to the question with all answers) concepts.

For Type 1 the best MRR results were obtained by using P2Q approach on combined concepts form CE and Categories (Cat1,2,3), thus semantic enrichment by CE can improve results. It should be noted also that I&I-STSS and LinSTSS are providing the same results for Cat1,2,3, because for all concepts form the categories we set their weight to the same value of 1, since those concepts are directly provided by the user (so according to their importance the highest weight is assigned). TFIDF and SN gave lower results as well as when combined with other sources. Also, it is interesting that I&I-STSS approach evaluated on TF-IDF concepts showed considerably better results than LinSTSS. The reason for this is that TFIDF and SN weights are lowering results because some words that are rear, and thus more weighted, may not be so important for the question. Also, for semantic enrichment SN considers only one word and not the whole text of the post, thus imposing higher error rate. The P@N results for this database with combined concepts form CE and Categories are shown in Fig. 1.

For Type 2 both for Answer and Thread concepts P2Q approach gave better results, except for user predefined Cat1,2,3 where I&I-STSS and LInSTSS performed a little bit better. Also, CE is providing now slightly lower results even in combination with user provided categories, so the best results are obtained using only Categories concepts. This may be because this database type contains only users who best answered at least ten questions, without questions they posted. Therefore, semantic analysis tools like CE cannot accurately extract concepts from the text in answer since it is usually a continuation of the question.

For Type 3 both for Answer and Thread concepts P2Q approach gave the best MRR results on combined concepts form CE and Categories. P@N results for combination of CE and Categories is shown in Fig 2. The highest P@30 result is obtained for Type 2 and 3 by using Categories concepts with the value of 90%, which is also the best result for the whole evaluation.

Finally, comparing the results achieved on each of the three databases we found that for profiling the user competence to give an answer for the provided question it is important not only to consider answers, which the user best answered (i.e. to profile knowledge), but also to take into account questions, which can express interests.

VI. Conclusion

In this paper we investigated the role of semantic similarity for each stage of IQRS process. For question and answer analysis we used semantic enrichment, TF-IDF and IQRS system features, and for their matching we experimented with three semantic similarity algorithms. Evaluation on subset of Yahoo! Answers L6 dataset showed that the best results are obtained by using semantic enrichment with proposed CE

[4]https://github.com/szitnik/InterestMining/blob/master/paper/inprogress_informatics2013/tex/informatics2013Results.pdf?raw=true

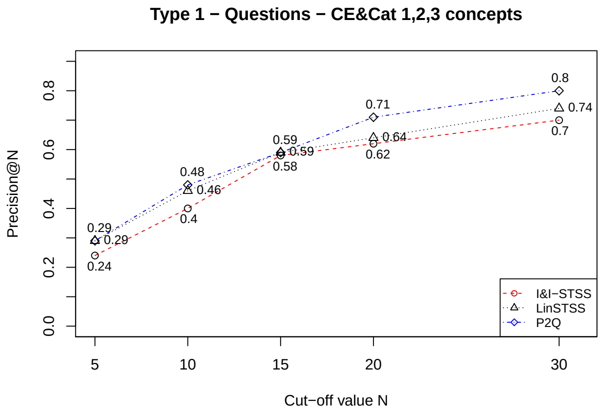

Figure 1: Precision@N results for database type 1 using questions and concepts from concept extractor and categories.

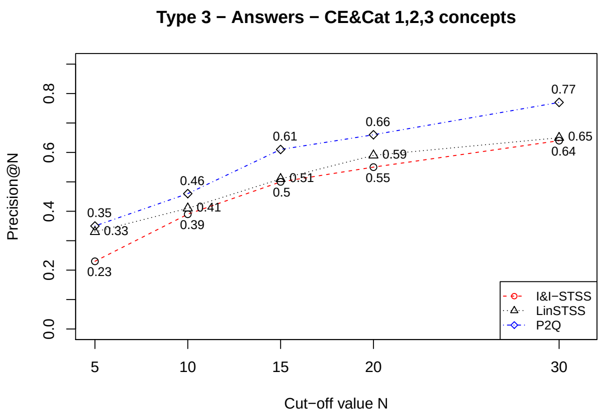

Figure 2: Precision@N results for database type 3 using answers and concepts from concept extractor and categories.

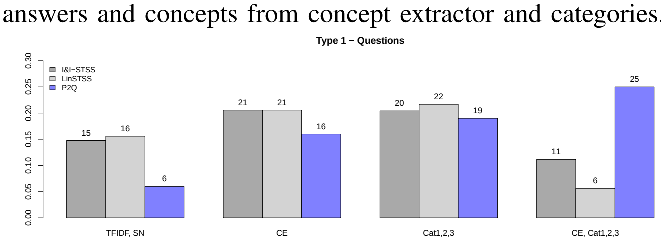

Figure 3: Mean reciprocal rank scores for database of type 1 on question concepts.

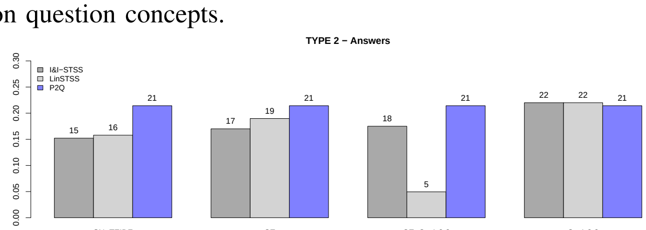

Figure 4: Mean reciprocal rank scores for database of type 2 on answers concepts.

approach and the introduced P2Q algorithm. Also, we found that for profiling the user competence to give an answer for the provided question it is important not only to consider answers which the user best answered (i.e. to profile knowledge), but also to take into account questions which can express interests.

In the future work we are going to incorporate the techniques from social network analysis for interest and knowledge prediction and together with methods for short text similarity investigate the impact of the new model.

Acknowledgment

The work has been supported by the Slovene Research Agency ARRS within the research program P2-0359 and part financed by the European Union, European Social Fund. It was also partially funded by the Ministry of Education and Science of the Republic of Serbia (projects III44009, 44006, 32047).

References

[1] B. Furlan, B. Nikolic, and V. Milutinovic, "A survey and evaluation of state-of-the-art intelligent question routing systems," *International Journal of Intelligent Systems*, vol. 28, no. 7, pp. 686–708, 2013.

[2] D. Lemire and O. Kaser, "Tagcloud drawing: Algorithms for cloud visualization," in *Proceedings of WWW 2007 Workshop on Tagging and Metadata for Social Information Organization*, 2007.

[3] M. A. Hearst and D. Rosner, "Tag clouds: Data analysis tool or social signaller?" in *Proceedings of the 41st Annual Hawaii International Conference on System Sciences*, 2008, pp. 160–160.

[4] B. Furlan, V. Batanović, and B. Nikolić, "Semantic similarity of short texts in languages with a deficient natural language processing support," *Decision Support Systems*, vol. 55, no. 3, 2013.

[5] B. Furlan, V. Sivački, D. Jovanović, and B. Nikolić, "Comparable evaluation of contemporary corpus-based and knowledge-based semantic similarity measures of short texts," *JITA-Journal of Information Technology and Applications*, vol. 1, no. 1, pp. 65–71, 2011.

[6] A. Islam and D. Inkpen, "Semantic text similarity using corpus-based word similarity and string similarity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 2, no. 2, pp. 1–25, 2008.

[7] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," in *Proceedings of the National Conference on Artificial Intelligence*, vol. 21, 2006, pp. 775–780.

[8] R. Speer and C. Havasi, "Representing general relational knowledge in conceptnet 5." in *Proceedings of Language Resources and Evaluation Conference*, 2012, pp. 3679–3686.

[9] F.-R. Chaumartin, "ANTELOPE - Une plateforme industrielle de traitement linguistique," *TAL : traitement automatique des langues : revue semestrielle de l'ATALA*, vol. 49, no. 2, pp. 43–71, 2008.

[10] H. Agt and R.-D. Kutsche, "Automated construction of a large semantic network of related terms for domain-specific modeling," in *Advanced Information Systems Engineering*, C. Salinesi, M. Norrie, and O. Pastor, Eds., vol. 7908. Springer Berlin Heidelberg, 2013, pp. 610–625.

[11] G. A. Miller, "Wordnet: A lexical database for english," *Communications of the ACM*, vol. 38, pp. 39–41, 1995.

[12] M. Surdeanu, M. Ciaramita, and H. Zaragoza, "Learning to rank answers on large online qa collections," in *In Proceedings of the 46th Annual Meeting for the Association for Computational Linguistics: Human Language Technologies*, 2008, pp. 719–727.