TEM Journal. Volume 6, Issue 2, Pages 315-325, ISSN 2217-8309, DOI: 10.18421/TEM62-17, May 2017.

Automatic Clustering of Papers in Thematic Fields and Working Sessions

Yordan Kalmukov

University of Ruse, 8 Studentska Str. ,7017 Ruse, Bulgaria

***Abstract* – This article proposes an approach for semi-automatic paper clustering that suggests: which papers to be grouped together in common working sessions, the number of sessions (thematic fields), and the inter-session and intra-session organization, i.e. the order of sessions and the order of papers within a session. In contrast to the existing solutions this approach does not try to group papers in a predefined number of equally-sized sessions, but follows the natural data fragmentation to achieve higher accuracy by producing integral and consistent clusters, independent on the number of target sessions and their size. Experimental results show that the proposed solution is feasible and provides very good initial grouping of papers that greatly facilitates the preparation of the conference program.**

***Keywords* - Automatic conference program generation; agglomerative hierarchical clustering; comparing clusterings.**

1. Introduction

Information technologies have become an important part in organizing scientific events during the last years. Dozens of specialized conference management systems (cms) have been developed that make the work of all participants involved in paper submission and review easier. As a web application, the conference management system proves both reliable and flexible service accessible anytime anywhere in the world. After the initial submission authors can easily track the status of their papers. Reviewers can evaluate papers at a convenient for them time anywhere in the world. But the highest benefit goes to conference organizers and chairs. They get a highly automated, logically centralized tool for both monitoring and management of all processes related to paper submission, assignment, reviewing and decision making.

One of the most important and challenging tasks the PC chairs should do is the assignment of reviewers to papers. The result of it directly impacts the quality of the conference and its public image. For highly selective conferences, having low acceptance ratio, it is crucial that every paper is evaluated by the most competent reviewers in its subject domains. Even a small inaccuracy in the assignment could significantly ruin authors' trust in that event. The assignment could be performed both manually and automatically. Manual assignment is feasible for conferences having a small amount of submitted papers. However, when the number of papers and reviewers increases the manual assignment gets less and less accurate due to the many constraints it should satisfy. For that reason all commercially available conference management systems offer an automatic assignment of reviewers to papers. It is usually based on explicit description of papers and reviewers' competences provided by both authors (during paper submission) and reviewers (during registration).

Another important and hard task to manually accomplish is preparing the conference program, more precisely clustering papers in thematic fields and working sessions. Papers related to the same subject domain should be ordered one after another in the same session (time slot). Correct grouping maximizes the number of experts in a specific subject domain within the individual sessions – a prerequisite for better and intensified discussions, and know-how transfer. Furthermore, it allows thematically-disjoint sessions to run in parallel while participants are still able to see all the presentations they are interested in.

As conference management systems rely on explicit description of papers and reviewers' competences for automatic assignment of reviewers

DOI: 10.18421/TEM62-17
https://dx.doi.org/10.18421/TEM62-17

Corresponding author: Yordan Kalmukov,
University of Ruse, 8 Studentska Str. ,7017 Ruse, Bulgaria
Email: JKalmukov@ecs.uni-ruse.bg

© 2017 Yordan Kalmukov; published by UIKTEN. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 License.
The article is published with Open Access at www.temjournal.com

to papers, the same explicit description could be also used to cluster papers in thematic fields and working sessions. Usually the implemented method of describing papers and competences suggests a similarity measure to calculate a *paper-to-reviewer* similarity between all submitted papers and registered reviewers. If the similarity measure is symmetric, it could be also used to calculate *paper-to-paper* similarity that can be subsequently converted to a distance. Then clustering could be done in the following ways:

- ***Manual***. PC chairs use the explicit description of papers, provided by the authors during the paper submission to cluster papers manually.
- ***Semi-automatic***. The clustering algorithm suggests which papers to be grouped together and in how many sessions, but the PC chairs can monitor and navigate the clustering process all the time. In semi-automatic mode, *PC chairs do not set up the number of working sessions in advance, but the algorithm gets the freedom to follow the natural data fragmentation so that papers are optimally clustered* independent on the number of target sessions and the number of papers within a session. Then the interim results of clustering are presented to the PC chairs and they can "tell" the algorithm to continue to divide large clusters to smaller ones and/or to combine smaller clusters to larger ones.
- ***Fully automatic***. PC chairs set up the number of working sessions, then the clustering algorithm groups the accepted papers automatically in the specified number of sessions precisely – not more, not less. Thus in fully automatic mode, the clustering algorithm does not have the freedom to follow the natural data division, but should group papers in exactly *n* clusters. If the specified number of clusters is too small, there is a significant chance that highly dissimilar papers are grouped together in a single session. And the opposite – if the number of specified clusters is too large, then papers related to the same subject domain will get highly fragmented and placed in different sessions.

Manual clustering is hard to accomplish, especially in case of a large number of accepted papers. The fully automatic mode is fast and comfortable, but cannot guarantee high accuracy. Semi-automatic clustering is the optimal grouping method – it provides both accuracy and high level of automation.

Independent on the degree of automation, manual clustering and user interface to reassign papers to different sessions should be always available – not because of inaccuracy in the clustering algorithms or the similarity measures, but because there is always a chance that authors chose incorrect keywords. Subjective factors should not be ignored.

2. Related work

Most commercially available conference management systems, including EasyChair [31], Microsoft Conference Management Toolkit [34], The MyReview System [36], OpenConf [35], EDAS: Editor's Assistant [32], Web Submission and Review Software by Shai Halevi [38], CyberChair [30], IAPR Commence [33], YaCOMAS [39], Web-based Conference Management Tool (WCMT) [37], ConfSys [28] and ConfTool [29] provide manual grouping of papers and a web interface to create and manage sessions, and to manually assign papers to them. No existing conference management system is known to offer an automatic clustering of papers in working sessions. The official documentation of the most utilized system, EasyChair [31], states that conference topics may be used in future to generate the conference program, but does not specify how exactly and when.

A few publications about clustering papers in working sessions exist in the scientific literature. Paul André et al. propose a multi-staged community-supported approach [1]. As a first stage program committee members, based on their expertise, perform an initial grouping of papers. Then during the second stage authors are asked to evaluate and refine the initial grouping by indicating which papers fit or do not fit in a common session with their paper. Finally, during the third stage, conference organizers resolve conflicts, fix incoherent sessions and produce the schedule. Some automation is also used for initial paper filtering (before stage one), so that PC members are asked to group papers just in areas they are competent in. This significantly reduces the number of papers a single PC member should group, i.e. simplifies his/her work a lot. As the proposed community-supported approach combines the expertise of many professionals, it may provide a very high clustering accuracy, that could be even higher than the traditional methods of manual sorting of papers by PC chairs. However, it exclusively relies on participation of the majority. André et al. tested their approach during the conference on human-computer interaction, CHI 2013. Results show that 30% of committee members took part in the first (committeesourcing) stage, while the authors of 87% of the accepted papers took part in the second (authorsourcing) stage. The higher participation of authors is somehow expected as they have an explicit interest that their papers are grouped with semantically similar ones. However, extensive participation of PC members is also important for creating an adequate initial grouping.

Tadej Škvorc et al. suggest a two-tier constrained clustering method for automatic conference scheduling that automatically assign paper presentations to predefined schedule slots [20]. It uses existing clustering algorithms to group papers in clusters based on similarities between papers. The latter are initially calculated by using the vector space text analysis model then augmented with the reviewers' preferences explicitly stated by the PC members during the bidding process. At the beginning, a feature (bag-of-words) vector, containing the title and the abstract, is built for every single paper. Then a td-idf weighting scheme is applied to these vectors to weight their components. Next, the co-bidding graph, that contains reviewers' bids, is converted to feature vectors by using the Personalized PageRank (PPR) algorithm. Two papers are connected in the co-bidding graph if the same reviewer stated his/her willing to review them both. Furthermore, graph edges are weighted in accordance with reviewers' bids. For example, if the reviewer stated "I want to review this paper" for both papers, then the corresponding edge will get higher weight than if the reviewer has selected "I can review it" for one of the papers. Finally, the PPR vectors are merged with their respective "tf-idf" vectors before calculating the similarities. Škvorc et al. tested their method during the 14-th conference on artificial intelligence in medicine, AIME 2013. According to their analysis the best results were obtained by K-means clustering algorithm, which consistently returned well-structured clusters [20]. Results also show that augmenting the vector space text analysis model with reviewers' bids/preferences increases accuracy (measured by silhouette score) by just 4.5%.

Tin Huynh et al. suggest that the similarity between two papers p_i and p_j could be calculated based on the number of common references they cite and the number of papers that cite them both together [9]. They improve the existing CCIDF algorithm [5], used by CiteSeer, by augmenting the co-reference network with co-citing network. The former is constructed by analyzing the "references" section of the papers while the latter is built upon papers citing both p_i and p_j simultaneously. Although the authors tested their CCIDF+ algorithm with quite small dataset, results show that it achieves a bit higher precision and recall in comparison with traditional CCIDF. In the context of paper clustering for automatic conference program generation however, the proposed CCIDF+ algorithm will not provide any benefit over traditional CCIDF since accepted papers are not cited yet.

3. Automatic clustering of papers in thematic fields and working sessions

This section proposes an approach for semi-automatic paper clustering that suggests:

- Which papers to be grouped together in common working sessions;
- The number of working sessions;
- The inter-session and intra-session organization, i.e. the order of sessions and the order of papers within a session. Clustering results indicate whether two or more sessions should be ordered sequentially one after another or they could run in parallel.

In contrast to existing solutions this approach does not try to group papers in a predefined number of equally-sized sessions, but follows the natural data fragmentation to achieve higher accuracy by producing integral and consistent clusters, independent on the number of target sessions and their size. If for example a cluster gets too large and cannot fit in a single time slot, the PC chair can split it to two sequential slots (sessions) while keeping the intra-session order of papers consistent and logical. Alternatively if a cluster is too small and cannot entirely occupy a single time slot then the approach suggests the PC chair which other cluster (session) to move to the same slot. The process is semi-automatic precisely because the PC chair should interact clustering at its final stage. This interaction however is not hard, nor time-consuming.

To group papers in thematic fields we first have to calculate semantic similarity between every pair of papers. Then an existing clustering algorithm could be applied over the similarity matrix to group papers in clusters. Calculation of similarity factors is strongly dependent on the chosen method of describing papers. If there is no explicit description then paper similarities could be calculated based on information retrieval techniques such as the vector space text analysis model or latent semantic analysis [6]. If authors have provided an additional explicit description of their papers (for example by selecting keywords from a predefined set), then similarities could be calculated by using an appropriate similarity measure – Jaccard's index, Dice's coefficient or other, depending on the method of describing papers.

According to [11] probably the most accurate explicit method of describing papers is by taxonomy of keywords. It comes with two important advantages:

- The implied hierarchical structure of the taxonomy allows similarity measures to take into account not only the number of exactly matching keywords, but in case of non-matching keywords to calculate how semantically close they are. Thus, if two papers do not share even a single keyword in common, a non-zero similarity factor could still be accurately calculated between them.
- The method requires minimum user participation and is usually implemented with no additional interactions. During paper submission, authors are required to select (from the conference taxonomy) all keywords that apply to their papers. And that's it. No additional actions are needed. As discussed earlier community-supported approaches may achieve higher accuracy but they are entirely dependent on participation of the majority of the community members. If the level of participation is low, papers' description may be sparse or inaccurate, leading to incorrect assignment of reviewers to papers or incorrect paper clustering.

Together with the method of describing papers and reviewers' competences, [11] proposes a symmetric similarity measure to calculate <paper–reviewer> similarities (1). It is based on the well-known and widely used Dice's coefficient, but in contrast to it, the suggested similarity measure could be also applied between sets whose elements are semantically related to each other (as concepts in taxonomy are). So it allows a non-zero similarity factor to be accurately calculated even if the two papers do not share any keyword in common [11]. The same similarity measure could be used to calculate <paper–paper> similarities as well.

$$Sim(p_i, p_j) = \frac{\sum_{k_m \in KWp_i} \max_{k_n \in KWp_j} (Sim(k_m, k_n)) + \sum_{k_n \in KWp_j} \max_{k_m \in KWp_i} (Sim(k_n, k_m))}{|KWp_i| + |KWp_j|} \tag{1}$$

where:

$Sim(p_i, p_j)$ – similarity between paper p_i and paper p_j.

k_m – the m-th keyword, describing paper p_i.

k_n – the n-th keyword, describing paper p_j.

KWp_i – set of keywords, describing the i-th paper.

KWp_j – set of keywords, describing the j-th paper.

$Sim(k_m, k_n)$ – semantic similarity between the m-th keyword describing p_i and the n-th keyword describing p_j. This is the similarity between concepts k_m and k_n in the taxonomy.

Similarity between two taxonomy concepts, i.e. $Sim(k_m, k_n)$, could be calculated by either edge counting similarity measures, such as the one proposed by Zhibiao Wu and Martha Palmer [23], or information-theoretic similarity measures such as the one of Dekang Lin [12]. We implemented Wu and Palmer's similarity measure (2). It is in fact the Dice's coefficient applied over the definition of the term "path", i.e. a set of edges – twice the number of common edges divided by the number of all edges (including duplicates).

$$Sim_{Wu\ \&\ Palmer}(k_m, k_n) = \frac{2 \times N_0}{2 \times N_0 + N_1 + N_2} \tag{2}$$

where:

N_0 – the distance (in number of edges) between the root and the closest common ancestor C_0 of the two concepts (C_m) and (C_n) whose similarity is being calculated.

N_1 – the distance from C_0 to one of the concepts, say C_m. C_m represents k_m – the m-th keyword describing the i-th paper.

N_2 – the distance from C_0 to the other concept – C_n. C_n is the node representing k_n – the n-th keyword describing the j-th paper.

If during the paper submission authors are able to specify the *level of applicability* of each selected keyword to the paper it describes, then these levels could be used to modify the calculated similarity between k_m and k_n. The levels of keywords applicability we use in our conference management system are: "High" – the keyword represents a major topic of the paper; "Medium" – the keyword represents a topic that is relatively important, but is not a major for the paper; "Low" – the keyword represents a topic that is of little importance to the paper, and "Not Applicable" – the keyword represent a topic that is not related to the paper at all.

The specified level of keyword applicability could be used as a weight that proportionally lowers the calculated similarity $Sim(k_m, k_n)$ if k_m is medium or low applicable to the paper p_i.

$$Sim_{weighted}(k_m, k_n) = Sim(k_m, k_n) \times w(k_m)$$

$$w(k_m) = \left\{ \begin{matrix} 1, & \text{if level = High} \\ 0.75, & \text{if level = Medium} \\ 0.25, & \text{if level = Low} \\ 0, & \text{if level = Not Applicable} \end{matrix} \right\} \tag{3}$$

Once similarities between papers are calculated, an existing clustering algorithm could be applied on the produced similarity matrix to group papers in thematic fields. Probably the most popular clustering algorithm is the *K-means clustering* [13],[7]. It groups *n* objects in *k* clusters so that every object is assigned to the cluster with the nearest center (mean). Each assignment changes the number of elements within a cluster, thus the position of its center. The new position is calculated in a way that it minimizes the sum of distance squares within the cluster. Although K-means is a great and easy to implement algorithm, I find it inappropriate for clustering papers, because it groups objects in a predefined number of clusters. In other words, the algorithm is unable (not allowed) to follow the natural data fragmentation and will try to group data in an inconsistent predefined structure. If the predefined number of clusters is too small, there is a significant chance that highly dissimilar papers are grouped together in a single session. If the specified number of clusters is too large, then papers related to the same subject domain will get highly fragmented and placed in different sessions. One may say that after the automatic clustering ends, PC chairs can merge smaller clusters or split larger ones. Yes, they can, but that will be a hard task as the K-means algorithm performs a “flat clustering”. Flat clustering creates a flat set of clusters without any explicit structure that would relate clusters (or cluster elements) to each other [14]. So if PC chairs have to merge or split clusters they need to do some additional analysis by themselves, because the algorithm cannot tell them which clusters to merge or which elements to pull out from a larger cluster.

To achieve more accurate and flexible clustering we need to know the inter-cluster and intra-cluster relationships. Having this information, PC chairs can easily merge or split clusters in order to fit them in the predefined conference schedule. Inter-cluster and intra-cluster relationships could be easily obtained by implementing an *agglomerative hierarchical clustering* [10],[8],[14]. It follows a “bottom-up” approach to build clusters. At the beginning, it treats each object (paper) as a separate cluster. Then iteratively, on each step, it merges the two nearest clusters together to form a larger one. It continues to do so until all objects are grouped in a single large cluster or until it is terminated. The key benefit of this algorithm is that when moving bottom-up it builds a weighted binary tree between clusters. Every edge connecting two clusters has an associated weight showing the semantic distance between these clusters. The generated binary tree is usually graphically represented as a coloured dendrogram (figure 1.). It gives us important information: which papers to be grouped together; in how many clusters (working sessions), how to order these clusters in the conference schedule, which sessions should be ordered sequentially one after another and which can run in parallel, how to order papers within sessions. This is key information that allows PC chairs to precisely tune the conference program after the clustering algorithm ends.

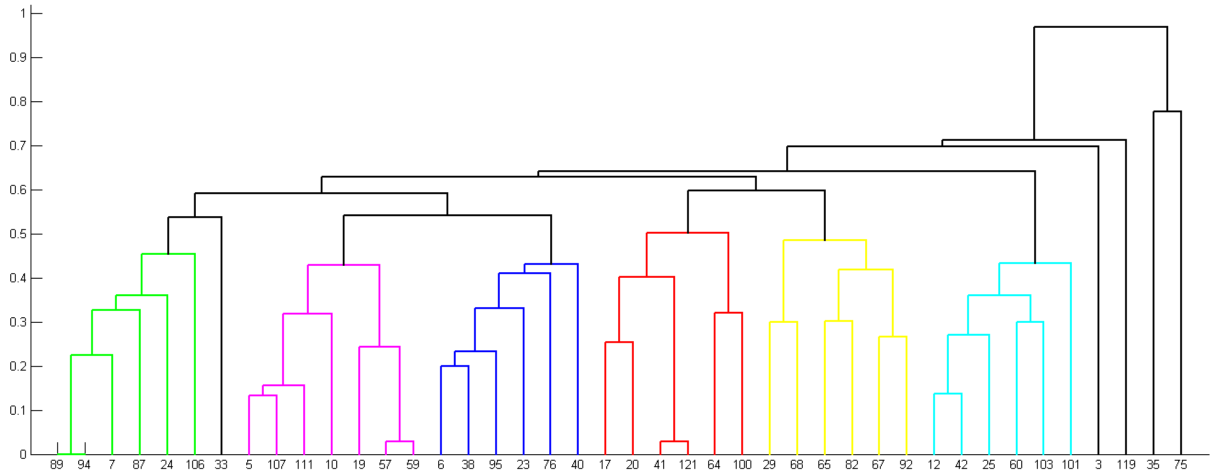

Figure 1. Coloured dendrogram built by an agglomerative hierarchical clustering. It shows clustering proposal for papers accepted to CompSysTech 2016 conference. X-axis represent paper identifiers while Y-axis – semantic distance between clusters (and papers).

As the clustering algorithm works with distances, rather than similarities, the calculated semantic similarities between papers should be converted to distances. The value of the similarity measure (1) varies in the range [0, 1], thus:

$$d(p_i, p_j) = 1 - Sim(p_i, p_j) \qquad (4)$$

where: $d\ (p_i, p_j)$ – semantic distance between paper p_i and paper p_j.

When moving bottom-up, on each step, the algorithm is merging the two "closest" clusters. Which clusters are closest depends on the chosen linkage method. As the calculated similarities between papers are single dimension values, the appropriate linkage methods should be single linkage clustering [19], complete linkage clustering [3] and average linkage clustering [16],[21]. Which one is best could be determined by the value of the *cophenetic correlation coefficient* [18],[22]. As seen in the dendrogram in figure 1., any two objects are connected to each other by a link that has a specific height. This height is called a *cophenetic distance* [22]. To evaluate how well the generated cluster tree reflects the original data we can compare the calculated cophenetic distances to the real semantic distances between the papers. If clustering is good, cophenetic distances should have a strong correlation with the real semantic distances. Experiments reveal that average linkage achieves highest results (cophenetic correlation of 0.85 for clustering papers of CompSysTech 2016 conference).

When using hierarchical clustering, PC chairs cannot specify the number of target clusters directly as they can do with K-means flat clustering. Instead they can control the number of clusters and their size by specifying a threshold value for the cophenetic distance. All links having a cophenetic distance higher than the threshold value are considered to be inconsistent, i.e. they are connecting unrelated clusters that should not be merged.

4. Experimental evaluation

A proposed clustering could be verified by both an internal and external evaluation. The internal evaluation checks how well the clustering represents the real input data, but ignores all subjective aspects. The cophenetic correlation coefficient is an example of an internal evaluation metric. It could be used to evaluate linkage methods or even entire clustering algorithms, but it cannot tell how good the clustering is in respect to human judgment. In contrast, the external evaluation relies on human experts to validate the clustering as a whole, or each of its cluster assignments. External evaluation is usually performed by using a reference clustering built by experts in the relevant subject domain.

To evaluate the accuracy of the produced clustering it will be compared to the real papers clustering that appear in the conference program of the last three editions of the CompSysTech conference [24] – 2016, 2015 and 2014. CompSysTech [24] is an international broad-area conference that covers all topics in computer system and information technologies, including hardware, software and their applications in other subject domains. Its conference proceedings are published by ACM and indexed in Scopus. All submitted papers are double-blindly reviewed by three independent reviewers who originate from countries different than those of the authors. The conference program of CompSysTech is prepared in several stages. First, the PC secretary performs a rough grouping of papers based on their subject domains. Then PC and OC chairs, mostly Prof. Boris Rachev and Prof. Angel Smrikarov, fix discrepancies and refine the clustering. Finally, the conference program is sent (in advance, before publishing) to the authors of all accepted papers to check for irrelevant grouping or incorrect order of papers. Due to this multistage process that involves many human experts we assume that grouping of papers in the conference program is optimal, thus we take it as a reference clustering.

There are some existing metrics that could be used to compare two clusterings, most of which are pair counting similarity measures. They take into account all pairs of objects and count the number of pairs on which the two clusterings agree or disagree. A pair of objects (papers) could fall into one of four cases:

TP – true positive. This is the case when two papers p_i and p_j are grouped together in a common cluster in the proposed clustering and they truly belong to one and the same cluster in the reference clustering, i.e. the algorithm correctly determined the relationship between p_i and p_j.

TN – true negative. Two papers p_i and p_j are assigned to different clusters in the proposed clustering and they truly belong to different clusters in the reference clustering, i.e. the algorithm correctly determined the relationship again.

FP – false positive. Two papers p_i and p_j are grouped together in a common cluster in the proposed clustering, but they actually belong to different clusters in the reference clustering.

FN – false negative. Two papers p_i and p_j are assigned to different clusters in the proposed clustering, but they actually belong to one and the same cluster in the reference clustering.

Knowing the number of both correctly and incorrectly identified pairs allows us to determine how close the produced clustering matches the reference one. This could be done by any of the existing clusterings comparing metrics, such as Rand's index (5) [17], Jaccard's index (6) [15], [2] or Fowlkes–Mallows Index (7) [4].

Rand's index:

$$RI = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

Jaccard's index:

$$JI = \frac{TP}{TP + FP + FN} \tag{6}$$

Fowlkes–Mallows index:

$$FMI = \sqrt{\frac{TP}{TP + FP} \cdot \frac{TP}{TP + FN}} \tag{7}$$

Rand's index represents the percentage of correctly identified pairs. However, as it accounts the number of true negatives, its value will be always high, so it is not very informative and precise. This could be easily noticed from the result tables. In contrast, Jaccard's index completely disregards true negatives thus provides higher precision. However, if a larger cluster is divided into two smaller ones in the reference clustering (due to a lack of enough time in a single time slot, for example) this could inaccurately lower the value of the Jaccard's index. The Fowlkes–Mallows index is the geometric mean of precision and recall. In addition to these metrics it is also reasonable to give the *percentage of correctly classified papers*. A paper is correctly classified if it is assigned it to the same cluster as it appears in the reference clustering. For example, if a paper is assigned by the algorithm to the "Software engineering" cluster and it truly belongs to the "Software engineering" session in the reference clustering, then the paper is correctly classified. Alternatively, if the paper is assigned to the "Software engineering" cluster, but it appears in another session in the reference clustering then it is incorrectly classified.

CompSysTech 2016

The conference program of CompSysTech'16 is organized in 9 thematic fields fitted in 9 sessions (time slots) which run in 3 main tracks in parallel. The program is available here [25]. Due to the lack of enough time within a single session, some larger thematic fields are split into two sequential sessions - one after another. This is the case with software engineering and artificial intelligence and its applications in medicine and localization. In the reference clustering, however "Software Engineering 1" and "Software Engineering 2" are combined since they actually form a single cluster that is unnaturally divided into two sessions. The same applies to "AI in Medicine" and "AI in Localization". Furthermore, the clustering algorithm could not make a reliable distinction between the last two as the taxonomy contains keywords about artificial intelligence, but not for its applications such as localization.

Two of the nine sessions (C1 and C2) are related to e-Learning. As authors submit their e-learning papers to a separate specialized track (Educational Aspects of Computer System and Technologies), these two sessions are objectively excluded from the experimental analysis, because their papers are processed separately from the others. Similarly, exclusion of the e-learning papers is done for the other two editions of CompSysTech (2015 and 2014) as well.

The grouping proposed by the agglomerative hierarchical clustering is illustrated in figure 1. in the form of a coloured dendrogram. It suggests that papers are grouped in 7 thematic fields and shows which papers to be grouped together in common sessions. The number of groups in the proposed clustering exactly corresponds to the number of sessions in the reference clustering, after exclusion of the two e-learning sessions.

It should be noted that the dendrogram in figure 1. is built without accounting the levels of keywords applicability. Thus all keywords selected to describe a specific paper have an equal weight in the description. As mentioned earlier during the paper submission, authors can indicate how much exactly each selected keyword applies (relates) to the paper they submit. Accounting the levels of applicability may increase clustering accuracy, because the algorithm will try go group together papers which are *highly related* to a specific topic, rather than papers described by many similar keywords which are actually not important to the description of the respective papers. Experimental results prove this assumption. Accounting the *levels of keywords applicability* makes the proposed clustering closer to the reference one. It could be seen in table 1. that all clustering comparison metrics agree with that.

Table 1. Results of comparing the proposed automatically-generated clustering of CompSysTech'16 to the real clustering of papers in the conference program

	Rand's index	Jaccard's index	Fowlkes–Mallows index	Correctly classified, %
CompSysTech 2016, the levels of keywords applicability are *not* accounted	0.89	0.42	0.59	73.81%
CompSysTech 2016, the levels of keywords applicability are accounted	0.91	0.51	0.67	78.57%

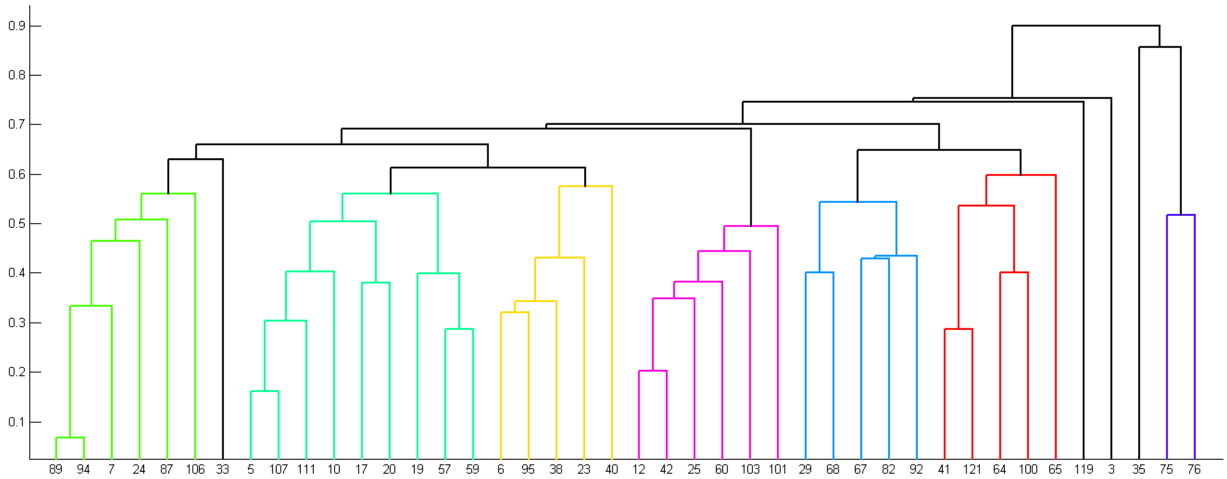

Figure 2. Clustering proposal for CompSysTech'16 after accounting the levels of keywords applicability

Accounting the levels of keywords applicability changes the clustering proposal. The new one is illustrated in figure 2. As seen in table 1. the new clustering proposal is closer to the reference clustering, although the similarity distances here are higher than those in figure 1. The latter is a result of lowering the calculated similarity in accordance with (3). However, the exact values of paper similarities (distances) are not important at all. What is important are their relationships as they determine proposed clustering.

A detailed analysis showed that no cluster in the proposed clustering actually corresponds to the thematic field "Software Security and Computer Networks" in the reference clustering. The latter includes just 4 papers. According to the clustering algorithm, 2 of them are related to software engineering and the other 2 to cloud computing and distributed systems. That is probably the reason why these 4 papers are fitted right after the "Software Engineering" session and before "Cloud Computing". So, if we assume that the reference clustering does not contain a separate group "Software Security and Computer Networks" and 2 of its papers go to software engineering and the other 2 - to cloud computing, then the proposed clustering gets tightly closer to the reference one (table 2.).

Table 2. Results of comparing the proposed automatic clustering of CompSysTech'16 to the real clustering in the conference program, after dissolving "Software Security and Computer Networks" session

	Rand's index	Jaccard's index	Fowlkes–Mallows index	Correctly classified, %
CompSysTech 2016, the levels of keywords applicability are *not* accounted	0.90	0.51	0.68	80.95%
CompSysTech 2016, the levels of keywords applicability are accounted	0.93	0.63	0.78	88.10%

CompSysTech 2015

The conference program of CompSysTech'15 [26] is organized in 8 thematic fields in 8 sessions running in 3 main tracks in parallel. There are two larger fields ("Software Engineering" and "Information Retrieval and Information Systems") that cannot fit within a single session so they are split to two sequential sessions placed one after another. The proposed clustering is presented in figure 3. The dendrogram shows very high level of fragmentation. There are just 2 or 3 groups of papers that tend to cluster, while the semantic distances between the other papers are quite high suggesting that they cannot be clustered. However, they should be clustered somehow. For example papers 4, 48, 28, 86, 30 and 95 are grouped together although the distance between some of them is higher than 0.6.

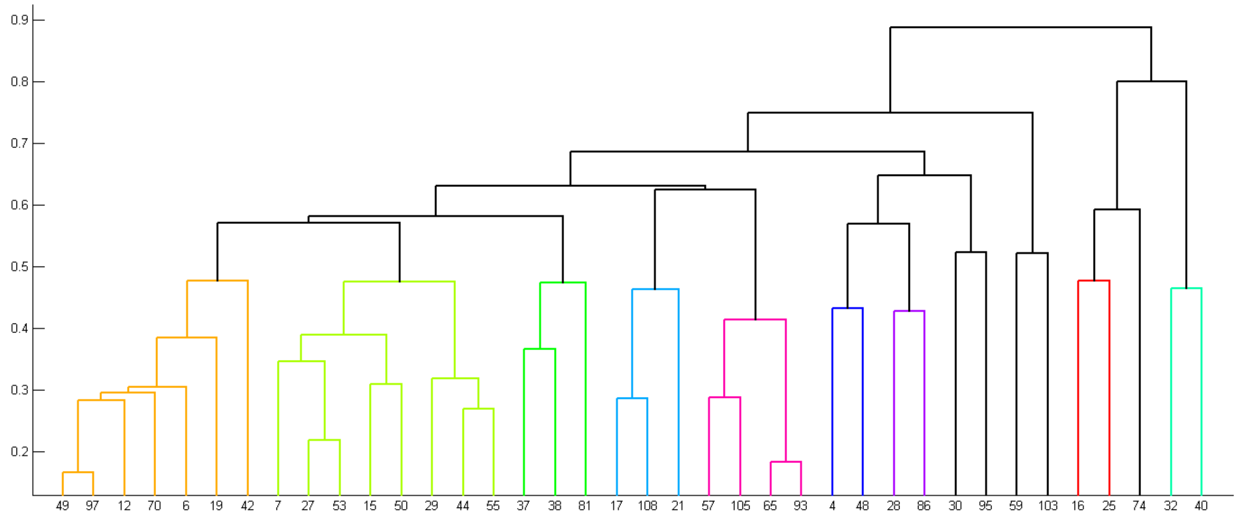

Figure 3. Proposal for clustering accepted papers of CompSysTech'15 conference

Due to the lack of well differentiated clusters we can expect that the proposed clustering does not much resemble the reference one. It is clearly shown by the clustering comparison metrics in table 3.

It is not easy to explain why such high level of fragmentation occurs. A detailed analysis of the selected keywords showed that some of the incorrectly classified papers are described by too many unrelated and not weighted keywords. That makes the focus of a paper unclear (for automatic processing). If a paper is described by many keywords they should be weighted, so that the author indicates which topics (keywords) are important and which are not. This information will help the algorithm to group together papers which are highly related to a specific topic, rather than papers which are described by many similar keywords. However, there are papers which are well described with 3-4 related or weighted keywords and they are still incorrectly classified. This is mostly because they concern topics different than those of the other papers, so they are indeed hard to be automatically classified.

Table 3. Results of comparing the proposed automatically-generated clustering of CompSysTech'15 to the real clustering of papers in the conference program

	Rand's index	Jaccard's index	Fowlkes–Mallows index	Correctly classified, %
CompSysTech 2015, the levels of keywords applicability are *not* accounted	0.89	0.35	0.53	73.68%
CompSysTech 2015, the levels of keywords applicability are *not* accounted Combined sessions IR 1 + IR 2; and SE 1 + SE 2	0.89	0.37	0.55	81.58%
CompSysTech 2015, the levels of keywords applicability are accounted	0.87	0.33	0.50	68.42%

CompSysTech 2014

The conference program of CompSysTech'14 [27] is organized in 8 thematic fields fitted in 8 sessions running in 4 main tracks in parallel. There is also a fifth parallel track, but it contains high-school and university students' papers which are not reviewed and not published. As in the other two editions of CompSysTech, e-learning papers (sessions G1 and G2) are excluded from this analysis as they are explicitly submitted to a separate track. From the rest 6 topics there is no one that contains more papers than a single time slot can handle. So there are no thematic fields split into several sequential sessions this time.

Results from the clustering comparison analysis are given in table 4. As in year 2016, they show that much higher clustering accuracy is achieved when the levels of keywords applicability are taken into account. Additionally, the proposed clustering can better resemble the conference program if the yellow cluster (papers 99, 105 and 113) is manually merged with the pink cluster. Both, the cophenetic distances in the dendrogram (figure 4.) and the conference program support this. Papers in the yellow cluster are about data mining and information retrieval just as the papers in the pink cluster. However, papers in the yellow cluster are described by keywords related just to information retrieval, while all papers in the pink cluster also include some keywords from artificial intelligence. This looks to be the most significant reason why the algorithm has formed two separate clusters.

Table 4. Results of comparing the proposed automatically-generated clustering of CompSysTech'14 to the real clustering of papers in the conference program

	Rand's index	Jaccard's index	Fowlkes–Mallows index	Correctly classified, %
CompSysTech 2014, the levels of keywords applicability are *not* accounted	0.86	0.37	0.54	74.36%
CompSysTech 2014, the levels of keywords applicability are accounted	0.91	0.52	0.68	82.05%
CompSysTech 2014, the levels of keywords applicability are accounted, the yellow cluster is merged with the pink one	0.92	0.58	0.73	84.62%

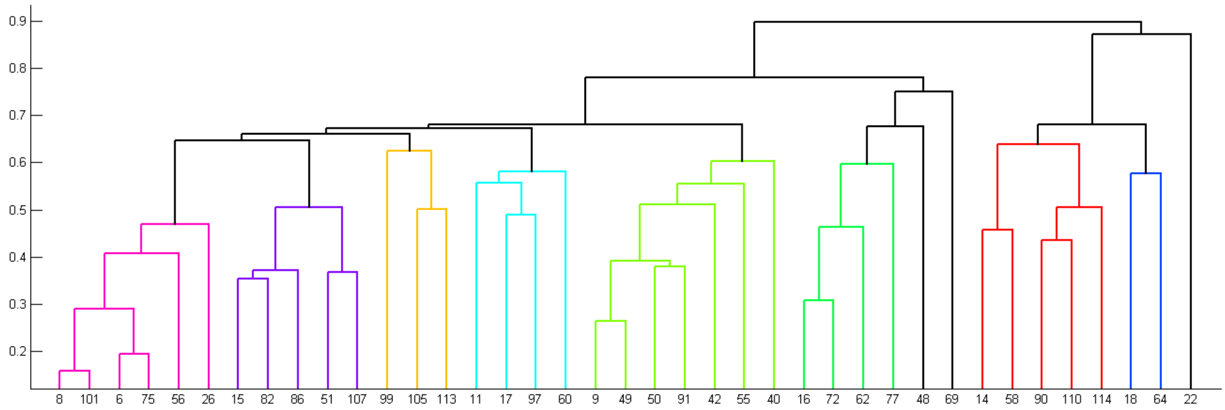

Figure 4. Proposal for clustering accepted papers of CompSysTech'14 conference. The levels of keywords applicability are taken into account.

5. Conclusion

Experimental results show that the proposed semi-automatic clustering approach is feasible and provides very good initial grouping of papers that greatly facilitates the preparation of the conference program. The produced clustering is far from perfect and does not exactly resemble the target clustering, but is close to it. Furthermore, it is much easier, when you have such a good initial start, to perform manually a couple of reassignments, rather than to start grouping papers from the scratch.

In addition, the experimental analysis showed that describing papers with many unrelated and not weighted keywords is a bad practice as it makes the focus of a paper unclear (for automatic processing). If a paper is described by many keywords, they should be mandatory weighted, so that the author indicates which topics are essential and which are not. This is an important observation that should be taken into account when designing and developing conference management systems. It applies not just when clustering accepted papers, but more importantly when assigning reviewers to the submitted papers.

References

[1]. André, P., Zhang, H., Kim, J., Chilton, L., Dow, S. P., & Miller, R. C. Community clustering: Leveraging an academic crowd to form coherent conference sessions. In First AAAI Conference on Human Computation and Crowdsourcing, November 2013

[2]. Ben-Hur, A., A. Elisseeff, I. Guyon, A stability based method for discovering structure in clustered data, in: Pacific Symposium on Biocomputing, 2002, pp. 6–17.

[3]. Defays, D. An efficient algorithm for a complete link method. The Computer Journal (British Computer Society) 20 (4): 364–366, 1977.

[4]. Fowlkes, E. B. & C. L. Mallows (1983), "A Method for Comparing Two Hierarchical Clusterings", Journal of the American Statistical Association 78, 553–569.

[5]. Giles, C. L., Bollacker, K. D., & Lawrence, S. CiteSeer: An automatic citation indexing system. In Proceedings of the third ACM conference on Digital libraries (pp. 89-98). ACM. May 1998.

[6]. Grossman, D., Frieder, O. Information Retrieval: Algorithms and Heuristics 2nd Ed. Springer, The Netherlands, 2004, ISBN: 1-4020-3004-5.

[7]. Hartigan, John A and Manchek AWong. Algorithm AS 136: A k-means clustering algorithm. Applied Statistics, pages 100–108, 1979.

[8]. Hastie, T., Tibshirani, R., Friedman, J. The Elements of Statistical Learning (2nd ed.). Springer New York, ISBN 0-387-84857-6.

[9]. Huynh, T., Hoang, K., Do, L., Tran, H., Luong, H., & Gauch, S. Scientific publication recommendations based on collaborative citation networks. In Collaboration Technologies and Systems (CTS), 2012 International Conference on (pp. 316-321). IEEE.

[10]. Johnson, Stephen C. Hierarchical clustering schemes. Psychometrika, 32(3):241–254, 1967.

[11]. Kalmukov, Y. Describing Papers and Reviewers' Competences by Taxonomy of Keywords. Computer Science and Information Systems, 2012, No 9(2), pp. 763-789, ISSN 1820-0214.

[12]. Lin, D. An Information-Theoretic Definition of Similarity. Proceedings of the Fifteenth International Conference on Machine Learning ICML '98, ISBN:1-55860-556-8, 1998.

[13]. MacQueen, J. B. Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, 1967, pp. 281–297. MR 0214227.

[14]. Manning, Christopher D., P. Raghavan and H. Schütze. Introduction to Information Retrieval. Cambridge University Press, 2008.

[15]. Meilă, M. Comparing clusterings—an information based distance. Journal of multivariate analysis, 98(5), 873-895, 2007.

[16]. Murtagh, F. Complexities of Hierarchic Clustering Algorithms: the state of the art. Computational Statistics Quarterly 1: 101–113, 1984.

[17]. Rand, W. M. (1971). "Objective criteria for the evaluation of clustering methods". Journal of the American Statistical Association. American Statistical Association. 66 (336): 846–850. doi:10.2307/2284239. JSTOR 2284239.

[18]. Rohlf, F., Fisher, D. Test for hierarchical structure in random data sets. Systematic Zool., 17:407-412, 1968.

[19]. Sibson, R. SLINK: an optimally efficient algorithm for the single-link cluster method. The Computer Journal (British Computer Society) 16 (1): 30–34, 1973.

[20]. Škvorc, T., Lavrač, N., & Robnik-Šikonja, M. Co-Bidding Graphs for Constrained Paper Clustering. In OASIcs-OpenAccess Series in Informatics (Vol. 51). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016

[21]. Sokal, R., Michener, C. A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin 38: 1409–1438, 1958.

[22]. Sokal, R., Rohlf, F. The comparison of dendrograms by objective methods. Taxon, 11:33-40, 1962.

[23]. Wu, Z., M. Palmer. Verb semantics and lexical selection. 32nd. Annual Meeting of the Association for Computational Linguistics (1994), pp. 133 -138.

[24]. CompSysTech – International Conference on Computer Systems and Technologies, http://www.compsystech.org

[25]. CompSysTech'16 – conference program, http://www.compsystech.org/docs/CST16-Programme-1.pdf

[26]. CompSysTech'15 – conference program, http://www.compsystech.org/docs/CST15-Programme.pdf

[27]. CompSysTech'14 – conference program, http://www.compsystech.org/docs/CST14-Programme.pdf

[28]. ConfSys Conference Management System, http://confsys.encs.concordia.ca/

[29]. ConfTool: Conference and Event Management Software, http://www.conftool.net/

[30]. CyberChair Conference Management System, http://www.borbala.com/cyberchair/

[31]. EasyChair Conference Management System, http://easychair.org/

[32]. EDAS: Editor's Assistant, https://edas.info/doc/

[33]. IAPR Commence Conference Management System, http://iaprcommence.sourceforge.net/

[34]. Microsoft Conference Management Toolkit, https://cmt.research.microsoft.com/cmt/

[35]. OpenConf Conference Management System, https://www.openconf.com/

[36]. The MyReview System (does not have an official site anymore), http://myreview.sourceforge.net/

[37]. Web-based Conference Management Tool (WCMT), http://wcmt.sourceforge.net/

[38]. Web Submission and Review Software by Shai Halevi, https://people.csail.mit.edu/shaih/websubrev/

[39]. YaCOMAS Conference Management System, http://yacomas.sourceforge.net/