Abstracts

Abstracts

Abstracts are short summaries so the reader does not need to read the whole paper to identify the main findings.

An often-used structure is:

- **Introduction to the problem.** (*What are we talking about?* and *Why do anything?*) We write the **minimum needed for the reader to understand the findings** in the paper. Why is it not a trivial problem? What's wrong with the current approach?
- **What is your solution to that problem?** Explain what is **new** in the paper.
- **How do you demonstrate it works?** Usually through experimental results or mathematical analysis.
- **What are the implications?** (*So what?*) Sometimes this is not needed—it may be obvious from context (e.g. if the implications are simply that we made software run faster).

(And, as always, different people have different preferences; we break the rules if it helps. Also: content and word-limit considerations.)

A good abstract structure:

> During specific real-world events, some users of microblogging platforms could provide exclusive information about those events. The identification of such prominent users depends on several factors such as the freshness and the relevance of their shared information. This work proposes a probabilistic model for the identification of prominent users in microblogs during specific events. The model is based on learning and classifying user behavior over time using Mixture of Gaussians Hidden Markov Models. A user is characterized by a temporal sequence of feature vectors describing his activities. The features computed at each time-stamp are designed to reflect both the on- and off-topic activities of users. To validate the efficacy of our proposed model, we have conducted experiments on data collected from Twitter during the Herault floods that have occurred in France. The achieved results show that learning the time-series of users' actions is better than learning just those actions without temporal information.
>
> — Bazid, et al., CIKM, 2015.

There's a clear structure. (Although, I would more specific detail to the demonstration part.)

> Personalized itinerary recommendation is a complex and time-consuming problem, due to the need to recommend popular attractions that are aligned to the interest preferences of a tourist, and to plan these attraction visits as an itinerary that has to be completed within a specific time limit. Furthermore, many existing itinerary recommendation systems do not automatically determine and consider queuing times at attractions in the recommended itinerary, which varies based on the time of visit to the attraction, e.g., longer queuing times at peak hours. To solve these challenges, we propose the PersQ algorithm for recommending personalized itineraries that take into consideration attraction popularity, user interests and queuing times. We also implement a framework that utilizes geo-tagged photos to derive attraction popularity, user interests and queuing times, which PersQ uses to recommend personalized and queue-aware itineraries. We demonstrate the effectiveness of PersQ in the context of five major theme parks, based on a Flickr dataset spanning nine years. Experimental results show that PersQ outperforms various state-of-the-art baselines, in terms of various queuing-time related metrics, itinerary popularity, user interest alignment, recall, precision and F1-score.
>
> — Lim, et al., SIGIR, 2017.

> Location models built on social media have been shown to be an important step toward understanding places in queries. Current search technology focuses on predicting broad regions such as cities. Hyperlocal scenarios are important because of the increasing prevalence of smartphones and mobile search and recommendation. Users expect the system to recognize their location and provide information about their immediate surroundings.
>
> In this work we propose an algorithm for constructing hyperlocal models of places that are as small as half a city block. We show that Dynamic Location Models (DLMs) are computationally efficient, and provide better estimates of the language models of hyperlocal places than the standard method of segmenting the globe into approximately equal grid squares. We evaluate the models using a repository of 25 million geotagged public images from Flickr. We show that the indexes produced by DLMs have a larger vocabulary, and smaller average document length than their fixed grid counterparts, for indexes with an equivalent number of locations. This produces location models that are more robust to retrieval parameters, and more accurate in predicting locations in text.
>
> — Murdock, SIGIR, 2014.

What's wrong with these abstract structures?

> Designing and implementing a quality matchmaking service for Multiplayer Online Games requires an extensive knowledge of the habits, behaviors and expectations of the players. Gathering and analyzing traces of real games offers insight on these matters, but game server providers are very protective of such data in order to deter possible reuse by the competition and to prevent cheating. We circumvented this issue by gathering public data from a League of Legends server (information over more than 28 million game sessions). In this paper, we present our database which is freely available online, and we detail the analysis and conclusions we draw from this data regarding the expected requirements for the matchmaking service.
>
> — Véron et al., NOSSDAV, 2014.

Here there's too much introductory material. There's very little about what the authors actually did.

There's no answer to: How do you demonstrate it works? (Except it's "freely available online".)

The part about implications basically says "see the paper".

> SQLite is a small-sized database engine largely used in embedded devices and local application software. The availability of portable devices, such as smartphones, has been extended over the recent years and has contributed to growing adaptation of SQLite. This implies a high likelihood of digital evidences acquired during forensic investigations to include SQLite database files. Where intentional deletion of sensitive data can be made by a suspect, forensic investigators need to recover deleted records in SQLite at the best possible. This study analyzes data management rules used by SQLite and the structure of deleted data in the system and in turn suggests a recovery tool of deleted data. Further, the study examines major SQLite suited software as it validates feasible possibility of deleted data recovery.
>
> — Jeon et al., Pers. Ubiquit. Comput., 2012.

There's no answer to: How do you demonstrate it works? They say "This study ... suggests a recovery tool of deleted data"; this implies they didn't actually demonstrate that the "recovery tool" works.

Why not implement and test it?

Abstracts

(Tenses)

Tenses (in the whole paper)

The standard tense usage in computer science:

- When referencing other authors' works, use *simple past tense*.

  Lei et al. (2014) proposed ...

- Otherwise, we mostly use *simple present tense*.

  We introduce a new approach ...
  This paper describes ...
  Experimental results indicate ...

  Sometimes is beneficial to use *continuous present tense*, e.g.:

  ... for processing a ...
  ... in developing a ...

- (If we use future tense, we're probably doing something wrong—we can't predict the future.)

In my experience, this is the most common approach. Other approaches are still grammatically correct, but are less common, and they're harder to read.

What can go wrong...

This sentence is grammatically correct:

> The experiments showed that the proposed approach outperformed the available methods on the CMU Mocap database.
>
> — Hu and Xie, CGI, 2017.

However, using past tense with "outperformed" carries an implication that it no longer outperforms the available methods!

Modifying the tense...

> Our practical experiments **have demonstrated** that Social Drive works efficiently with low battery consumption and low networking overhead on popular mobile devices.
>
> — Hu et al., DIVANet, 2013.

Replace "have demonstrated" with "demonstrate".

> Our experimental results **have shown** that the new algorithm substantially improves the precision of traditional collaborative filtering algorithms.
>
> — Ding et al., ADC2006, 2006.

Replace "have shown" with "show".

Are these tenses suitable?

> We will demonstrate the system using Twitter data on a computer cluster.
>
> — Jia et al., SIGSPATIAL, 2016.

No! It's future tense. It should begin "We demonstrate ..." in simple present tense.

> In this paper, we propose an algorithm to partition both the search space and the database for the parallel mining of frequent closed itemsets in large databases.
>
> — Tang et al., ACM Southeast Conference, 2005.

Yes, it's simple present tense.

Are these tenses suitable?

> By exploiting prior knowledge about the data, the proposed technique generally outperforms PCA and its variants in similarity searches.
>
> — Megalooikonomou et al., CIKM, 2004.

Tricky! The word “proposed” is used as an adjective. We can use it in sentences of various tenses:

XYZ is the proposed technique.
XYZ was the proposed technique, but we retracted it.
XYZ will be the proposed technique, when we're done.

The adjective “proposed” does not determine the tense of the sentence. However, when its used as a verb, it is past tense:

We proposed XYZ.

In the example, the verbs “exploiting” and “outperforms” are in present tense.

Are these tenses suitable?

> We have implemented our techniques in MYSQL, which can provide built-in keyword-search capabilities using SQL.
>
> — Li et al., VLDB J., 2011.

“We have implemented...” is past tense. It's simpler to use present tense “We implement...”.
(Also we should delete the vague word “can”—does it “provide built-in keyword-search capabilities using SQL” or not?) (Also: “MySQL”.)

> Recently, Wang et al. proposed a privacy preserving public auditing mechanism for shared cloud data with supporting group dynamic.
>
> — Chen et al., MISNC, 2017.

Yes, simple past tense is suitable for citing prior work. (It would benefit from using compound adjectives, though.)

Abstracts

(Active writing)

> **Use the active voice. The active voice is usually more direct and vigorous than the passive** ...
>
> — *Strunk & White, Section III.11*

As simply as possible:

- “**Y** is **done** by **X**” or “**Y** is **done**” is passive because **Y** either comes before the agent **X** or the agent **X** is omitted.
- The active voice is “**X does Y**”, where the agent comes before the action.

For example:

> **We introduce** a **vector representation called diffusion curve textures** for mapping diffusion curve images (DCI) onto arbitrary surfaces

Writing actively should be the “default”—we write passively it whenever it helps.

Example:

> However, caches make it significantly harder to compute the worst-case execution time (WCET) of a task. To alleviate this problem, **cache locking has been proposed.**
>
> — Zheng et al., ACM T. Arch. Code Opt. (2009).

Here, we fix the tense:

> However, caches make it significantly harder to compute the worst-case execution time (WCET) of a task. To alleviate this problem, **cache locking is proposed.**

But it's *passive*. We change it to the active voice:

> However, caches make it significantly harder to compute the worst-case execution time (WCET) of a task. To alleviate this problem, **we propose cache locking.**

We use an agent ("we"), and we write "we do X" (not "X is done").

Abstracts

(Introduction)

The introduction part of the abstract naturally comes first. It answers:

- **Topic.** What are we talking about?
- **Motivation.** Why do we care? (Prior work misses something?)
- **Context.** What does the paper relate to? (External to the paper.)

These will naturally vary from paper to paper. They also tend to blend into one another.

A good example:

> In web search today, a user types a few keywords which are then matched against a large collection of unstructured web pages. This leaves a lot to be desired for when the best answer to a query is contained in structured data stores and/or when the user includes some structural semantics in the query. ...
>
> — Paparizos et al., SIGMOD, 2009.

Topic: Web search with structured queries with semantics.
Motivation: Improve beyond current unstructured web pages.
Context: Current search engines consider unstructured web pages.

> The availability of large data centers with tens of thousands of servers has led to the popular adoption of massive parallelism for data analysis on large datasets. Several query languages exist for running queries on massively parallel architectures, some based on the MapReduce infrastructure, others using proprietary implementations. Motivated by this trend, this paper analyzes the parallel complexity of conjunctive queries. ...
>
> — Koutris and Suciu, PODS, 2011.

Topic: Complexity of parallelism for analyzing large datasets.
Motivation: Many query languages; which is best?
Context: Used in large data centers.

The last sentence is both part of the introduction, but also helps answer *What is your solution to that problem?* (An example of the components blending into one another.)

> In today's large-scale data centers, energy costs (i.e., the electricity bill) are projected to outgrow that of hardware. Despite a long history of research in energy-saving techniques, especially low-power hardware, little work has been done to improve the power efficiency of data management software. Power-aware computing research at the application level has been found to be synergistic to that at the hardware and OS levels because it can provide more opportunities for energy reduction in the underlying systems. ...
>
> — Xu, SIGMOD, 2010.

Topic: The "power efficiency of data management software".
Motivation: To reduce data-center energy costs.
Context: Currently, data centers are large, and energy costs are high.

This abstract skips the introduction entirely:

> In this paper, we propose a fingerprinting solution to protect valuable numeric relational data from illegal duplications and redistributions. We introduce a twice-embedding scheme. In the first embedding process, we embed a unique fingerprint to identify each recipient to whom the relational data is distributed. The embedding process is controlled by a secret key. Meanwhile, the fingerprint can be detected using the same secret key to prove ownership at a numerical confidence level. The second embedding process is designed for verifying the extracted fingerprint and giving a numerical confidence level. Thus, once a suspect copy is found, numerical confidence level can be provided both to identify the owner and the illegal distributor. The experiment shows that our solution is effective and robust to various attacks.
>
> — Guo, Wang, and Li, SAC, 2006.

We can deduce that some kind of data is being protected by fingerprinting. There is **no motivation** (so we don't know why it's important to do this). There's **no context** (so we don't know how this relates to anything else).

> In data applications such as information integration, there can be limited access patterns to relations, i.e., binding patterns require values to be specified for certain attributes in order to retrieve data from a relation. As a consequence, we cannot retrieve all tuples from these relations. In this article we study the problem of computing the *complete* answer to a query, i.e., the answer that could be computed if all the tuples could be retrieved. A query is *stable* if for any instance of the relations in the query, its complete answer can be computed using the access patterns permitted by the relations. We study the problem of testing stability of various classes of queries, including conjunctive queries, unions of conjunctive queries, and conjunctive queries with arithmetic comparisons. We give algorithms and complexity results for these classes of queries. We show that stability of datalog programs is undecidable, and give a sufficient condition for stability of datalog queries. Finally, we study data-dependent computability of the complete answer to a nonstable query, and propose a decision tree for guiding the process to compute the complete answer.
>
> — Li, VLDB J., 2003.

Too much technical detail in the introduction. Not the **minimum needed for the reader to understand the findings** in the paper.

Abstracts

(Solutions)

A good example description of the solutions component:

> Electromigration (EM) can cause severe reliability issues in contemporary integrated circuits. For the emerging three-dimensional integrated circuits (3D ICs), the introduction of through-silicon vias (TSVs) as the vertical signal carrier complicates the electromigration analysis. In particular, an accurate EM analysis on TSV arrays that are used in the power supply network is critical since the large current going through those TSVs can accelerate their degradation. In this work, we propose a novel EM analysis framework that focuses on TSV arrays in the power supply network, under the circumstance of uneven current distribution. The impacts of various design factors on the EM lifetime are discussed in detail. Our results reveal that the predicted TSV array lifetime is largely biased without proper current distribution analysis, resulting in an unexpected early failure.
>
> — Zou, et al., GLSVLSI, 2014.

If the reader wants the technical details (and they probably don't), they can read the paper. (I'm not keen on "under the circumstance" though.)

> Route planning for a set of locations based on trajectory searching is a hot topic. To obtain previous drivers' knowledge on route selection, some existing works search trajectories which are spatially close to the query locations. However, these trajectories may be only close to partial query locations or go to other locations beyond the query location set, which lead the algorithm to a poor performance. In this paper, we study a new model called *Route Planning for Locations Based on Trajectory Segment* (RPBTS). Given a set of ordered query locations, in order to pass as close as to the query locations, we plan a route by combining some intersecting trajectory segments. A greedy algorithm is employed to retrieve the optimal combinations which doesn't contain the two undesirable conditions mentioned above. To enhance the performance of the algorithm, we construct a regional landmark graph, where a regional landmark is a road segment frequently traversed by drivers in a certain region. Based on this graph, it's very likely a query location can be converted to a regional landmark. Finally, we propose a Random Selection strategy to further improve the efficiency. The effectiveness of our method is verified by empirical study based on a real trajectory data set.
>
> — Jiang et al., UrbanGIS16, 2016.

We don't need the details about every optimization.

From the previous slide:

> ... In this paper, we study a new model called *Route Planning for Locations Based on Trajectory Segment* (RPBTS). Given a set of ordered query locations, in order to pass as close as to the query locations, we plan a route by combining some intersecting trajectory segments. A greedy algorithm is employed to retrieve the optimal combinations which doesn't contain the two undesirable conditions mentioned above. To enhance the performance of the algorithm, we construct a regional landmark graph, where a regional landmark is a road segment frequently traversed by drivers in a certain region. Based on this graph, it's very likely a query location can be converted to a regional landmark. Finally, we propose a Random Selection strategy to further improve the efficiency. ...

Extracting what is important:

> ... We propose *Route Planning for Locations Based on Trajectory Segments*, where we plan a route that passes close to the query locations by combining intersecting trajectory segments. We use a greedy algorithm to retrieve the optimal combinations, and optimize the model using a regional landmark graph and a random selection strategy. ...

Abstracts

(Demonstration)

When describing in the abstract how we demonstrate the proposed solution works, we want to be **specific**, but we should avoid **too much detail**.

Not specific enough:

> Experimental results using real-life social networks show the effectiveness of our proposed simulation method.
>
> — Zeng et al., AWC, 2013.

It's not quantitative! (I.e., it has no numbers!) In fact, this sentence doesn't logically imply the method is good.

- *Good*: "real-life social networks" (although, maybe "real-world" is better).
- *Poor*: Does not describe the experiments.
- *Poor*: Does not explain how effectiveness is measured.

This is more specific, but still **not specific enough**:

> We evaluate our 3D-SWIFT and the results show that 3D-SWIFT can achieve 87.6% performance improvement compared to the state-of-the-art Wide IO.
>
> — Zhang et al., GLSVLSI, 2014.

First, we optimize the sentence:

> Experimental results indicate that 3D-SWIFT achieves up to 87.6% better performance compared to the state-of-the-art Wide IO.

- *Good*: specific about the measurement: "87.6%".
- *Good*: specific about what it's compared to: "state-of-the-art Wide IO".
- *Poor*: Does not explain what "performance" means.
- *Poor*: Does not describe the experiments.
- *Poor*: Does not describe the test data.

This one?

> ... Through evaluations of two major cellular networks on three maps of different scales and locations, we show that NetNavi can significantly improve network access quality at low cost for mobile users, increasing the average download throughput by 67.5% for 67.6% source-destination pairs with only 11.4% extra travel delay ~~(and as significant as 193.8% with only 3% extra travel delay in some cases)~~. ...
>
> — Shi and Xie, MobiArch, 2013.

Not bad, but let's get rid of the salesmanship.

- *Good*: specific about an example measurement: "67.5%".
- *Good*: specific about how performance is measured: "average download throughput".
- *Good*: specific about the test set: "two major cellular networks on three maps of different scales and locations".
- *Poor*: does not say what it's compared to, but we can infer it'll be whatever is state of the art (so it's not a big problem).

Abstracts

(Implications)

Most of the time, writing about implications is not necessary.

- *Reason 1*: We do X. The implication is that others can do X too. (Obvious!)
- *Reason 2*: The implications are often part of the motivation in the introduction.
- *Reason 3*: A reader of a paper on ABC is probably aware of the implications.

Many papers omit this part. And it would be a mistake to include it unnecessarily.

A good example of when we use it:

> Our approach provides an opportunity for ~~the~~ robots to understand ~~the~~ human instructions on a large scale.
>
> — Xie and Chen, WI and IAT, 2014.

Here, we have **a <u>new</u> capability** because of the research—it's important to recognize that we can do something now that we couldn't do before.

Another good example of discussing implications:

> ... The achieved results show that learning the time-series of users' actions is better than learning just those actions without temporal information.
>
> — Bazid, et al., CIKM, 2015.

Again, we have **a <u>new</u> capability**: we can use the <u>new</u> method XYZ to better perform ABC.