To improve the real-time processing of a collection of crowd-sourced data, consider the following suggestions:

1. Data Streaming Architecture: Implement a data streaming architecture that allows for continuous ingestion, processing, and analysis of crowd-sourced data. Use technologies such as Apache Kafka or Apache Flink to handle high volumes of data in real-time.

2. Scalable Infrastructure: Ensure that your infrastructure is scalable to handle increasing data volumes and concurrent user activity. Leverage cloud services like AWS, Azure, or Google Cloud Platform, which offer scalable solutions like serverless computing and auto-scaling capabilities.

3. Data Validation and Cleaning: Implement robust validation and cleaning mechanisms to ensure the quality and integrity of the crowd-sourced data. Apply filters, rules, and machine learning techniques to identify and remove erroneous or irrelevant data points.

4. Real-time Analytics: Utilize real-time analytics tools and frameworks to gain immediate insights from the crowd-sourced data. Implement streaming analytics solutions like Apache Spark Streaming or Elasticsearch to perform real-time aggregations, anomaly detection, or pattern recognition.

5. Distributed Computing: Leverage distributed computing frameworks, such as Apache Hadoop or Apache Spark, to process the crowd-sourced data in parallel across multiple nodes or clusters. This enables faster processing and analysis of large datasets.

6. Machine Learning and AI: Apply machine learning and artificial intelligence techniques to extract valuable information from the crowd-sourced data. Develop models for sentiment analysis, predictive analytics, or recommendation systems to derive insights and enhance the user experience.

7. Real-time Visualization: Implement real-time visualization tools and dashboards to provide stakeholders with instant access to the analyzed crowd-sourced data. Tools like Grafana, Kibana, or Tableau can help create interactive visualizations that enable better understanding and decision-making.

8. Security and Privacy: Pay utmost attention to security and privacy aspects while handling crowd-sourced data. Implement robust authentication, encryption, and access control mechanisms to protect sensitive information. Comply with relevant data protection regulations, such as GDPR or CCPA.

9. Continuous Monitoring and Optimization: Regularly monitor the performance and scalability of your real-time data processing system. Use monitoring tools like Prometheus or Datadog to track system metrics, identify bottlenecks, and optimize resource allocation.

10. Feedback Loop and User Engagement: Establish a feedback loop with the crowd-source data contributors to improve data quality and user engagement. Provide mechanisms for users to provide feedback, report issues, or suggest improvements. Actively engage with the crowd-source community to foster collaboration and enhance the overall data collection process.